

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Klemen Dolinšek

# **Spletni portal slovenske akvakulture**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mira Trebar

Ljubljana, 2016



Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

*Besedilo je oblikovano z urejevalnikom besedil  $\text{\LaTeX}$ .*



Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Posredovanje informacij preko spleta je povezano z različnimi pristopi in rešitvami, ki se uporabljajo za vnos podatkov. Kandidat naj v diplomski nalogi analizira razvoj prilagodljivih spletnih strani s programsko opremo SilverStripe. Predstavi in izdela naj spletni portal slovenske akvakulture, ki zagotavlja celovit pregled področja na spletu. Rešitev naj omogoča tudi enostavno in uporabniku prijazno urejanje spletnih strani za partnerje, ki so vključeni v bazo slovenske akvakulture in želijo predstaviti podatke o svojem podjetju.



## IZJAVA O AVTORSTVU ZAKLJUČNEGA DELA

Spodaj podpisani Klemen Dolinšek, avtor zaključnega dela z naslovom:

*Spletni portal slovenske akvakulture* (angl. *Web portal for Slovene aquaculture*)

### IZJAVLJAM

1. da sem pisno zaključno delo študija izdelal samostojno pod mentorstvom doc. dr. Mire Trebar;
2. da je tiskana oblika pisnega zaključnega dela študija istovetna elektronski obliki pisnega zaključnega dela študija;
3. da sem pridobil/-a vsa potrebna dovoljenja za uporabo podatkov in avtorskih del v pisnem zaključnem delu študija in jih v pisnem zaključnem delu študija jasno označil/-a;
4. da sem pri pripravi pisnega zaključnega dela študija ravnal/-a v skladu z etičnimi načeli in, kjer je to potrebno, za raziskavo pridobil/-a soglasje etične komisije;
5. soglašam, da se elektronska oblika pisnega zaključnega dela študija uporabi za preverjanje podobnosti vsebine z drugimi deli s programsko opremo za preverjanje podobnosti vsebine, ki je povezana s študijskim informacijskim sistemom članice;
6. da na UL neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve avtorskega dela v elektronski obliki, pravico reproduciranja ter pravico dajanja pisnega zaključnega dela študija na voljo javnosti na svetovnem spletu preko Repozitorija UL;
7. dovoljujem objavo svojih osebnih podatkov, ki so navedeni v pisnem zaključnem delu študija in tej izjavi, skupaj z objavo pisnega zaključnega dela študija.

V Ljubljani, dne 14. marca 2016

Podpis študenta/-ke:





*Rad bi se zahvalil mentorici doc. dr. Miri Trebar za vso strokovno pomoč ter nasvete pri načrtovanju in izvedbi spletnega portala.*

*Posebna zahvala pa gre tudi moji puncici Leni za moralno podporo in strpnost ob izdelavi diplomske naloge.*



# Kazalo

Povzetek

Abstract

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Pregled področja</b>	<b>3</b>
2.1	Arhitektura MVC . . . . .	3
2.2	Tehnologije . . . . .	5
2.3	Orodja . . . . .	10
<b>3</b>	<b>Načrtovanje</b>	<b>13</b>
3.1	Akvakultura . . . . .	13
3.2	Spletni portal . . . . .	14
<b>4</b>	<b>Implementacija</b>	<b>19</b>
4.1	Podatkovni model . . . . .	19
4.2	Rešitev v SilverStripe . . . . .	21
<b>5</b>	<b>Portal slovenske akvakulture</b>	<b>25</b>
5.1	Administrativni vmesnik . . . . .	25
5.2	Uporabniški dostop . . . . .	38
<b>6</b>	<b>Sklepne ugotovitve</b>	<b>45</b>
	<b>Literatura</b>	<b>47</b>



# Seznam uporabljenih kratic

kratica	angleško	slovensko
<b>CSS</b>	Cascading Style Sheets	Kaskadne stilske predloge
<b>HTML</b>	HyperText Markup Language	Označevalni jezik za izdelavo spletnih strani
<b>MVC</b>	Model-View-Controller	arhitekturni vzorec
<b>ORM</b>	Object-relational mapping	relacijska preslikavo objektov
<b>PHP</b>	Hypertext Preprocessor	programski jezik PHP
<b>RDBMS</b>	Relational database management system	sistem za upravljanje relacijskih baz podatkov
<b>SQL</b>	Structure Query Language	Strukturiran poizvedbeni jezik
<b>W3C</b>	World Wide Web Consortium	konzorcij nadzor in upravljanje standardov kot sta jezik HTML in jezik CSS



# Povzetek

**Naslov:** Spletni portal slovenske akvakulture

Diplomsko delo obravnava celovit pregled informacij na področju akvakulture v Sloveniji. Glavni cilj je bil izdelava predstavitvenega portala, ki na enem mestu povezuje ribogojnice in školjčičišča, s katerimi večinoma upravljajo manjša družinska podjetja. Spletna aplikacija omogoča prikaz splošnih in drugih informacij ter daje možnost vključevanja podjetij tako, da si lahko urejajo svoje spletno mesto s podatki, ki jih želijo posredovati svojim potrošnikom. Zasnovana je s programsko opremo SilverStripe v označevalnem jeziku HTML in vključuje prilagodljivo oblikovanje s predlogami CSS tako, da je pregled vsebin prilagojen tudi za vse pametne naprave. V implementaciji predlaganega spletnega portala so bili vključeni realni podatki o akvakulturi. Za urejanje samostojne spletne strani pa je kot partner predstavljena ribogojnica podjetja Fonda.si d.o.o.

**Ključne besede:** akvakultura, ribogojstvo, školjkarstvo, spletni portal, MVC, SilverStripe.





# Abstract

**Title:** Web portal for Slovene aquaculture

The thesis deals with a comprehensive review of information on a field of aquaculture in Slovenia. The main objective was the development of presentation portal which links fish farms and shellfish farms on one site, which are mainly managed by small family businesses. Web application displays general and other informations, and allows companies to show their data as they wish to provide it to their consumers by allowing them to edit it. It is designed with SilverStripe software, with markup language HTML and includes CSS to support responsive web design. The view adapts content to be shown on all smart devices. The implementation of the proposed web portal included real aquaculture data. Fish farming company Fonda.si d.o.o. was used for demonstration of independent web page.

**Keywords:** aquaculture, fish farming, shellfish farming, web portal, MVC, SilverStripe.



# Poglavje 1

## Uvod

Splošno poznavanje pojma akvakulture je pomanjkljivo in velikokrat narobe razumljeno, zaradi tega pa lahko prihaja do napačnih odločitev in neizkoriščenosti potencialov. Pogosto se pojem zamenjuje s pojmom ribogojstvo, kateri je le ožji pojem akvakulture, ne vključuje pa drugih vrst gojenja vodnih organizmov, od alg do vretenčarjev. Področje akvakulture je relativno novo in je najhitreje razvijajoča se panoga v pridobivanju hrane.

V nalogi smo se osredotočili na posredovanje informacij in s tem povečanje prepoznavnosti panoge. Skušali smo zagotoviti pregled celotnega področja akvakulture, ter hkrati omogočiti vsem ponudnikom predstavitev podjetja in svojih produktov.

Analizirali smo razvoj prilagodljivih spletnih strani, sistem pravic uporabnikov, ter hrambo podatkov. Želeli smo zagotoviti enostavno in uporabnikom prijazno urejanje spletnih strani za partnerje, kateri so vključeni v našo bazo slovenske akvakulture. Spletna aplikacija je bila izdelana s programskim orodjem SilverStripe in drugimi jeziki na področju spletnih tehnologij.



# Poglavje 2

## Pregled področja

Zaradi slabe osveščenosti ljudi o pomenu akvakulture in slabe prisotnosti ribogojnic na spletu, smo se odločili izdelati spletni portal, ki bo to dvoje povezal in predstavil širši javnosti. Na podlagi teh informacij smo določili arhitekturni model, izbrali tehnologije in orodija za izdelavo spletnega portala akvakulture.

### 2.1 Arhitektura MVC

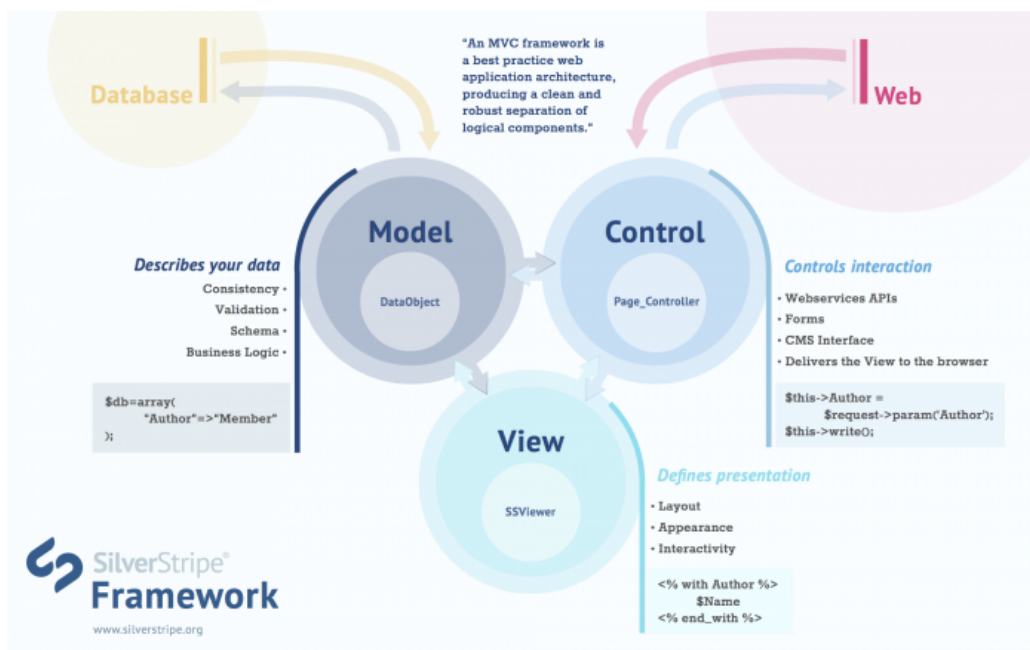
MVC (*Model-View-Controller*) predstavlja sodobno programsko zasnovo model-pogled-krmilnik. Gre se za arhitekturni vzorec, ki razdeli aplikacijo na tri med seboj odvisne komponente:

- Model (angl. *Model*): skrbi za konsistenco podatkov, validacijo podatkov, podatkovno shemo ter poslovno logiko
- Pogled (angl. *View*): določa predstavitev podatkov končnemu uporabniku.
- Krmilnik (angl. *Controller*): skrbi za interakcijo uporabnika z aplikacijo.

S tem pristopom lahko zmanjšamo kompleksnost razvoja aplikacije, saj dobimo z upoštevanjem arhitekturnega vzorca višjo preglednost nad datotečno

strukturo in programsko kodo.

### 2.1.1 SilverStripe MVC



Slika 2.1: Arhitektura model-pogled-krmilnik v ogrodju SilverStripe [1]

Ogrodje SilverStripe implementira arhitekturo MVC (Slika 2.1) v programskem jeziku PHP. Ponuja nam dva tipa naprednih razredov, katera lahko razširimo za potrebe spletne aplikacije. Prvi razred, obsega samo model in se imenuje podatkovni objekt (angl. *Data Object*), drugi pa vsebuje model, pogled in krmilnik imenovan stran (angl. *Page*). V ogrodju lahko krmilnik neposredno dostopa do podatkov iz modela, tako z dovoljenjem za branje in pisanje. Pogled lahko dostopa do podatkov krmilnika in modela, privzeto le z bralnim dostopom, medtem ko model lahko dostopa le do krmilnika. Vrsta dostopa med komponentami je odvisna od primerov oz. zahtev spletne strani. Ker imajo strani tudi svoj krmilnik, to pomeni, da so pripravljene za prikaz na uporabniškemu delu spletne strani. Imeti morajo tako tudi svoj

segment spletnega naslova (angl. *URL segment*). To je unikatni niz, s katerim ogrodje ugotovi, katero stran naložiti, prav tako pa poskrbi, da gradimo spletnim iskalnikom prijazen spletni naslov.

Ko uporabnik odpre spletni naslov, kateri je del SilverStripe, se najprej požene datoteka "main.php", v kateri se naložijo vse nastavitve ter inicializira okolje SilverStripe (angl. *SS Environment*), nato pa podatke preda razredu direktor (angl. *Director*). Ta razred nato razbije prejeti spletni naslov na manjše koščke – segmente spletnega naslova, ter od prvega proti zadnjemu išče zadnji segment, ki še ustreza zadetku med stranmi v podatkovni bazi. Tako ve, katero spletno stran je potrebno pokazati. Vsi deli naslova pa ni nujno da predstavljajo stran. Vsaka stran ima poleg svojega segmenta, lahko tudi akcijo (angl. *Action*), kar v kodi predstavlja klic določene funkcije. V tem primeru lahko funkcija spremeni (angl. *Override*) privzeti pogled, ter mu posreduje druge podatke, lahko pa tudi drugo predlogo.

## 2.2 Tehnologije

Spletni portal je narejen v odprtokodnem ogrodju SilverStripe, ki je napisan v jeziku PHP. Za urejanje podatkov uporabljamo istoimenski urejevalnik spletnih vsebin, pri čemer za shranjevanje uporabljamo podatkovno bazo (v našem primeru MySQL). Končne podatke nam SilverStripe posreduje v obliki jezika HTML, vse skupaj pa je oblikovano z dodatkom kaskadnih predlog (angl. *Cascading Style Sheets*). S 3. verzijo kaskadnih predlog in HTML verzije 5, smo sestavili stran, ki se zna prilagajati napravam z različno velikimi zasloni. Taki izdelavi spletne pravimo prilagodljiva oblika spletne strani (angl. *Responsive web design*).

### 2.2.1 HTML

Označevalni jezik HTML (angl. *HyperText Markup Language*) se uporablja za izdelavo spletnih strani. Poleg jezika CSS in jezika JavaScript, predstavlja temeljno tehnologijo za izdelavo spletnih strani, uporabniških vmesnikov za

spletne aplikacije in uporabniške vmesnike za mobilne aplikacije. Poleg značk za predstavitev besedil, kot so naslov, odstavek, seznam, povezava, citat, itd., jezik HTML omogoča tudi vključitev slik, objektov, skript kot na primer JavaScript, ter kaskadnih stilskih predlog. Osnova je datoteka tipa .html, katero znajo spletni brskalniki pridobiti od strežnika in pretvoriti v vizualne ali zvočne spletne strani. Jezik HTML semantično opiše strukturo spletne strani, skupaj z iztočnicami za predstavitev [6].

Leta 1980 je HTML zasnoval in predstavil fizik Tim Berners-Lee, z namenom delitve in uporabe dokumentov. Kasneje istega leta, je napisal prvi spletni brskalnik in strežniški program. Leta 1991 je bil objavljen prvi javni dokument, opis značk v jeziku HTML, kateri je vseboval 18 elementov, 11 od teh je še vedno obstajalo v verziji HTML 4. Danes je skrbnik za standard jezika HTML W3C (angl. *World Wide Web Consortium*) konzorcij. [7]

Opisni jezik je napisan v obliki elementov HTML, kateri so sestavljeni iz značk (angl. *Tags*), ti pa so zaprti s trikotnimi oklepaji. Brskalniki teh značk ne izpisujejo, temveč jih pretvorijo v vsebino.

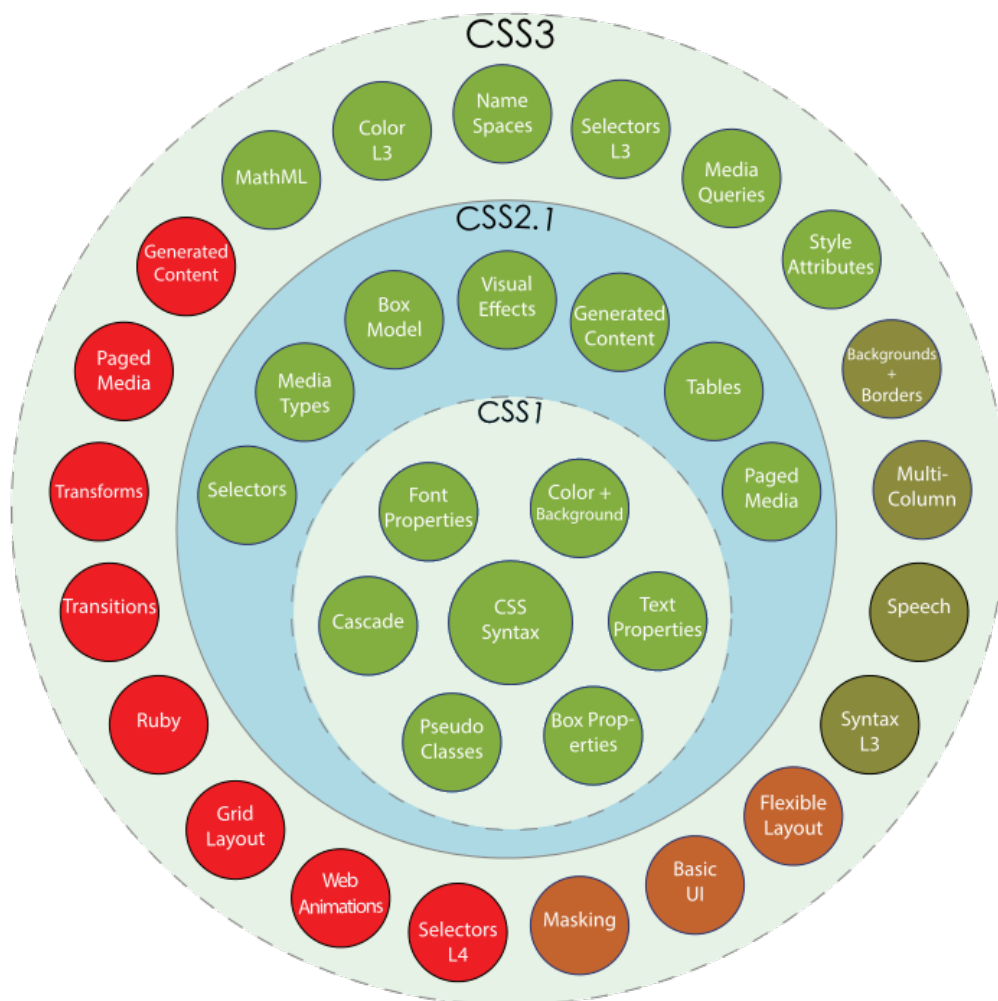
### 2.2.2 CSS

Kaskadne stilske predloge (angl. *Cascading Style Sheets*) so oblika slogovnega jezika, skrbijo pa za predstavitev spletnih strani. Z njimi definiramo stil elementov, kako naj se ti prikažejo na strani.

Pred uporabo CSS so bili skoraj vsi prezentacijski atributi HTML dokumentov vsebovani v HTML kodi; vse barve besedil, elementi ozadja, poravnave, okvirji in velikosti so bili opisani eksplicitno, vse večkrat ponovljeno. CSS omogoča avtorjem, da definirajo in shranijo te informacije v drugo datoteko, kar naredi HTML kodo bolj enostavno za pregled, prav tako pa se lahko izognemo večkratnemu ponavljanju enakega stila za prikaz.

Specifikacija in podpora za vzdrževanje CSS jezika je v rokah svetovnega konzorcija za splet (angl. *W3C*) (Slika 2.2) [4] [7].





Slika 2.2: Zgodovina podpore funkcij jezika CSS, od verzije 1, do verzije 3.[7]

### 2.2.3 JavaScript

”JavaScript” je visokonivojski in dinamičen programski jezik, ki se v interpretira v brskalniku. Poleg HTML in CSS je JavaScript osnovna tehnologija spletnih strani, katera je vključena v večini spletnih strani in je podprta pri vseh modernih spletnih brskalnikih. Sintaksa je v določeni meri podobna programskemu jeziku C, s pomanjkljivostjo vhodno-izhodnih operacij. Je zelo razširjena pri dinamičnih spletnih straneh, kamor jo lahko vgradimo ali pa

vključimo. Poleg spleta, pa je JavaScript uporabljen tudi v drugih okoljih, kot na primer v PDF dokumentih, ali pa kar v samem operacijskem sistemu.

### 2.2.4 PHP

PHP je odprtokodni programski jezik, ki se večinoma uporablja v strežniških okoljih za razvoj dinamičnih spletnih aplikacij. S svojim programskim jezikom je najbolj podoben jezikoma C in Java, sodi pa med šibko tipizirane programske jezike (angl. *weakly typed languages*). To pomeni, da jezik dovoljuje izvajanje katerekoli operacije na katerikoli podatkom. Ta lastnost je hkrati lahko tudi pomanjkljivost, saj lahko hitro manj veščim programerjem dopusti napako, hkrati pa oteži razhroščevanje aplikacije. [2]

Jezik PHP je lahko vključen v kodo HTML, ali pa ga uporabimo v kombinaciji z mnogimi sistemi za predloge, urejevalniki vsebin in spletnimi ogrodji. Kodo PHP običajno tolmači tolmač PHP, ki je vključen kot modul ali kot izvršljiv program. Interpretativno izvajanje programa pa pomeni tudi počasnejše izvajanje in manj možnosti optimizacij. Prednost takega izvajanja pa je enostavna prenosljivost programov, sprotno izvajanje programov (angl. *On the fly*), ter enostavna implementacija.

### 2.2.5 MySQL

SQL ali strukturirani poizvedbeni jezik [5], je poseben namenski programski jezik, namenjen upravljanju podatkov v sistemu za upravljanje podatkovnih baz (angl. *RDBMS*). MySQL je tak sistem, ki je namenjen upravljanju podatkov. Napisan je v programskem jeziku C in C++, za razčlenjevalnik uporablja program *YACC*, medtem ko za leksikalno analizo uporablja svoj algoritem. MySQL deluje na večini operacijskih sistemih, kar mu omogoča široko uporabo, ne glede na okolje [3].

V jeziku SQL, za operacije na podatki uporabljamo poizvedbe SQL. Med poizvedbami ločimo več vrst:

- izbirne poizvedbe:

- najbolj pogost stavek v jeziku SQL je SELECT za pridobivanje podatkov iz tabel
- manipulativne poizvedbe
  - Stavek INSERT: vnese podatke v podatkovno tabelo
  - Stavek UPDATE: posodobi obstoječe podatke
  - Stavek DELETE: izbriše podatke
- transakcijske poizvedbe
  - omogočajo kontrolo nad izvedbo poizvedb SQL, v kolikor poizvedba ni bila uspešna, se lahko vrnemo na prejšnje stanje relacijske baze (pred začetkom transakcije)

### 2.2.6 SilverStripe

Programska oprema SilverStripe sestoji iz intuitivnega sistema za upravljanje vsebin, imenovanega CMS (angl. *Content Management System*) in fleksibilnega ogrodja (angl. *Framework*) za izdelavo spletnih strani in spletnih aplikacij. Tako ogrodje kot tudi sistem za upravljanje vsebin sta izdelana v jeziku PHP, za svoje delovanje pa potrebujeta podatkovno bazo, kamor se shranjuje vsebina, ter spletni strežnik, na katerem je spletna stran prikazana. V skladu z modernimi smernicami, nudi ogrodje SilverStripe arhitekturo model-pogled-krmilnik (angl. *Model-View-Controller*), zagotavlja pa tudi relacijsko preslikavo objektov (angl. *Object-relational mapping*) in s tem omogoča razvijalcem hiter razvoj in nadaljnjo vzdrževanje kode, ki lahko skrbi za kompleksne spletne aplikacije [1].

Odprtokodno ogrodje in urejevalnik vsebin SilverStripe razvija istoimensko podjetje, ampak zaradi odprtokodnosti k razvoju prispevajo tudi drugi uporabniki. Programska oprema SilverStripe daje razvijalcem prostost, da naredijo kar želijo – pišejo dobro kodo, v elegantnem ogrodju. Razvoj kompleksnih spletnih aplikacij lahko naredimo hitro in robustno, brez omejitev

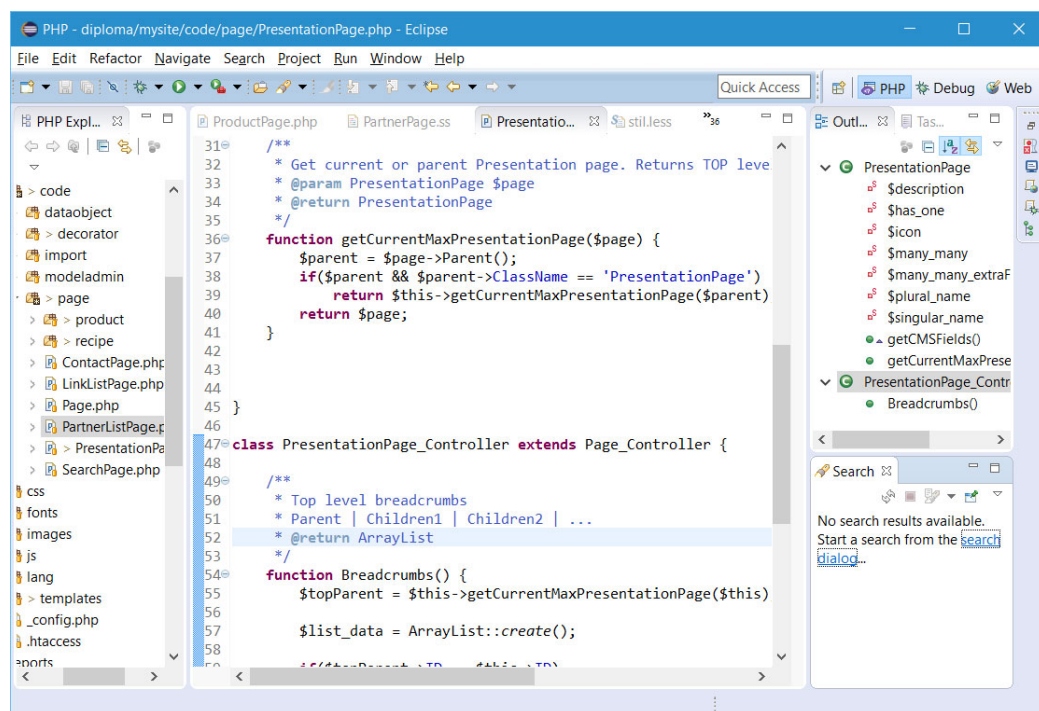
grafičnega vmesnika, funkcionalnosti pa lahko razširimo še z dodatnimi dodatki (angl. *Add-ons*). Ogrodje nas ne ovira niti pri razvoju predstavitvene spletne aplikacije pri uporabi HTML, CSS in JavaScript, kar omogoča oblikovalcev, da po kratkem času pregleda kode pričnejo z delom.

Programska oprema SilverStripe je razvita in deljena z BSD licenco, kar pomeni, da lahko s tako programsko kodo naredimo skoraj vse. Je odprta za javnost, kjer vsak lahko prispeva svoj delež. Razvoj, popravke in predloge upravljajo z sistemom za nadzor različic Git, na spletni strani [www.github.com/silverstripe](http://www.github.com/silverstripe).

## 2.3 Orodja

Pri implementaciji spletne aplikacije smo uporabili sistem za nadzor različic Git, z namenom lažjega sledenja delu in možnostjo varnostnega beleženja kopij verzij strani. Spletni portal je naložen na strežnik, kateri teče na sistemu *Ubuntu*, s spletnim strežnikom *Apache*. Lokalni je potekal v Windows okolju, s spletnim strežnikom *IIS* (angl. *Internet Information Services*). Na obeh strežnikih je nameščen tudi tolmač jezika PHP, verzije 5.5, z dodano podporo za povezovanje in upravljanje s podatkovno bazo MySQL.

Za pisanje programske kode je bil uporabljen program *Eclipse* (Slika 2.3). To je integrirano razvojno okolje (angl. *Integrated development environment*), katero vključuje osnovno delovno okolje (angl. *Workspace*), ter obsežen sistem dodatkov (angl. *plug-in*) za prilagoditev okolja. V našem primeru, smo uporabili okolje *Eclipse PDT for PHP*, katero doda funkcionalnosti primerne za razvoj PHP aplikacij. Za razhroščevanje spletne aplikacije, je bil vključen še modul *Xdebug*, kateri poveže tolmač PHP z razvojnim okolje Eclipse in tako omogoči pregled izvajanja kode.



Slika 2.3: Integrirano razvojno orodje Eclipse.



## Poglavje 3

# Načrtovanje

Kaj je akvakultura in kaj so njeni problemi? O naravnih problemih ne bomo govorili, bomo pa o problemih razpoznavnosti, ter o našem načrtu, kako reševati te probleme.

### 3.1 Akvakultura

Geografski raznolikosti Slovenije, lahko pripišemo tudi raznolikost akvakulture [8] [9]. Na obalnem delu ima Slovenija razvito marikulturo, a so zaradi pritiska z drugimi rabami morja, možnosti tu skoraj izčrpane. V panonskem svetu prevladujejo toplovodni ribniki, kateri dopuščajo vzgojo toplovodne akvakulture, medtem ko v predalpskem in kraškem delu Slovenije prevladujejo celinske vode, katere omogočajo različne oblike hladnovodne pridelave rib.

#### 3.1.1 Sladkovodna akvakultura

Ker se v Sloveniji v sladkih vodah vzreja praktično le ribe, lahko govorimo kar o ribogojstvu. Na tem področju lahko rečemo, da z izjemo nekaj večjih ribogojnic, prevladujejo večinoma manjša družinska podjetja, ki z relativno majhno povprečno proizvodnjo pokrivajo potrebe lokalnega trga.

### 3.1.2 Marikultura

Ob 46 km dolgi obali, kjer morje dosega v povprečju globino 20 metrov, so pred dobrimi tremi desetletji posamezniki začeli s poskusno vzreje školjk klapavic. Ker so bili rezultati dobri, se panoga razvija naprej. Na podlagi zakona, so določili tri območja za gojenje školjk in eno območje za gojenje morskih rib, v skupni velikosti 1.180.000 m<sup>2</sup> [8]. Ta območja so porazdeljena na parcele, katerim s koncesijo podelijo vodne pravice. Primer dobre prakse morske ribogojnice je družinsko podjetje Fonda.si d.o.o. v Piranskem zalivu in je trenutno edina morska ribogojnica v Sloveniji.

## 3.2 Spletni portal

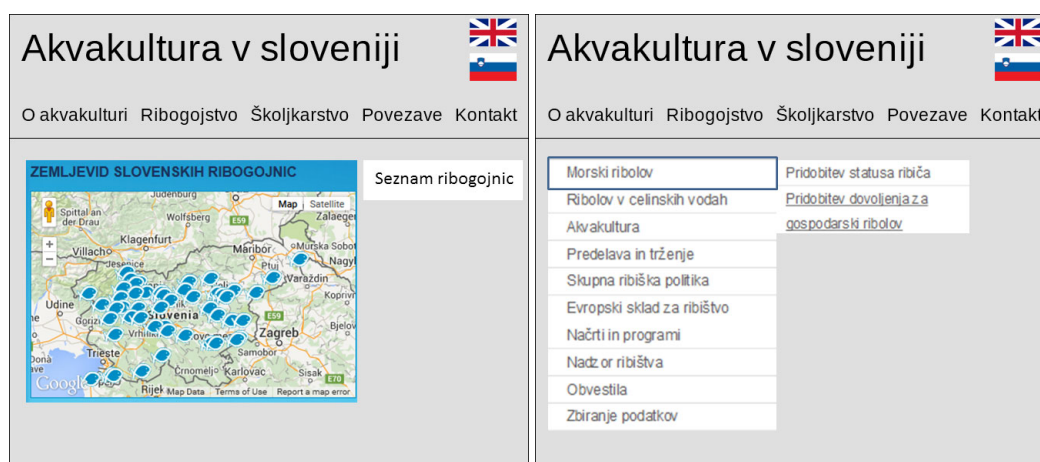
Namen spletnega portala je predstaviti akvakulturo in združiti podjetja, ki se ukvarjajo z ribogojstvom in školjkarstvom v Sloveniji, na enem mestu in povečati njihovo razpoznavnost. Glede na trenutne smernice pri izdelavi spletnih aplikacij, smo se odločili za sodobno rešitev, ki bo omogočala prikaz vsebin na vseh napravah. Poleg slovenskega jezika, bo predstavitevna stran še v enem tujem jeziku, to je angleščina. Spletni portal mora omogočati tudi enostaven pregled in iskanje podatkov.

Njegovo načrtovanje in implementacija sta razdeljena na:

- Programski model: V tem delu smo pretvorili potrebe spletne strani v razrede, kateri bodo poskrbeli za shrambo podatkov.
- Uporabniški vmesnik: V tem delu smo upoštevali vse predpostavke iz načrtovanja in se odločili kako jih vključili v predlogo.

Na podlagi tega, smo naredili skelet (angl. *Wireframe*), za lažjo predstavo kaj vključiti v aplikacijo in se lotili oblike strani (Slika 3.1).





(a) Seznam partnerjev z zemljevidom

(b) Strani s povezavami



(c) Podatki o ribogojnici

(d) Produkti v akvakulturi

Slika 3.1: Skeletni modeli za razmestitev elementov na strani

Predstavitvena stran je razdeljena na tri glavne dele:

- glava: na voljo imamo glavno navigacijo za preklap med stranmi, možnost za preklap med slovenskim in angleškim jezikom, ter sekundarno navigacijo, skupaj z iskalnikom. Vse to se zgradi dinamično, glede na podatke v podatkovni bazi.
- vsebina: prikazuje vsebino, glede na izbrano stran v glavi.
- noga: je skrita, ob kliku na simbol puščice na sredini strani, se nam

odpre prostor s statičnimi podatki o avtorju, mentorici in fakulteti.

Z glavno navigacijo se lahko pomikamo med glavnimi stranmi in s tem vplivamo na izpis in prikaz sekundarne navigacije in vsebine. Vsebina kot tudi njena postavitev se torej spreminjata glede na izbrano podstran. Uporabljamo več različnih postavitev strani, kot npr: enostavna spletna stran z možnostjo besedila in slik, stran s seznamom partnerjev, do kompleksnejših podstrani, s podatki partnerja in njegovimi produkti. Nekateri tipi vsebin lahko imajo tudi terciarno navigacijo – zavihke. S tem smo dosegli bolj pregledno porazdelitev tam, kjer je veliko različne vsebine za prikaz na eni strani, hkrati pa je sekundarna navigacija že uporabljena za drugo funkcionalnost.

## Uporabniki

Zaradi potreb urejevalnika vsebin SilverStripe, imamo za urejanje portala več skupin uporabnikov (slika 3.2):

- administrator: posebna skupina z vsemi pravicami
- partner: skupina za zunanje uporabnike, katera lahko ureja le svoje podatke in podatke svojih produktov
- urednik vsebin: skupina za zunanje uporabnike, katera lahko ureja le predstavitvene strani

Administrator je posebna skupina, katera ima omogočen dostop do vmesnika, z namenom urejanja vseh vsebin spletnega portala. Omogočen ima popolni pregled z možnostjo urejanje vseh vsebin in povezav, ki jih vključuje spletni portal. Le administrator ima možnost, da lahko doda novega uporabnika v sistem in mu določi skupino.

Za celovito rešitev urejanja spletnega portala neodvisno od administratorja, smo dodali skupini uporabnikov, katerima smo določili omejene pravice in so namenjene partnerjem, ter urednikom akvakulture.

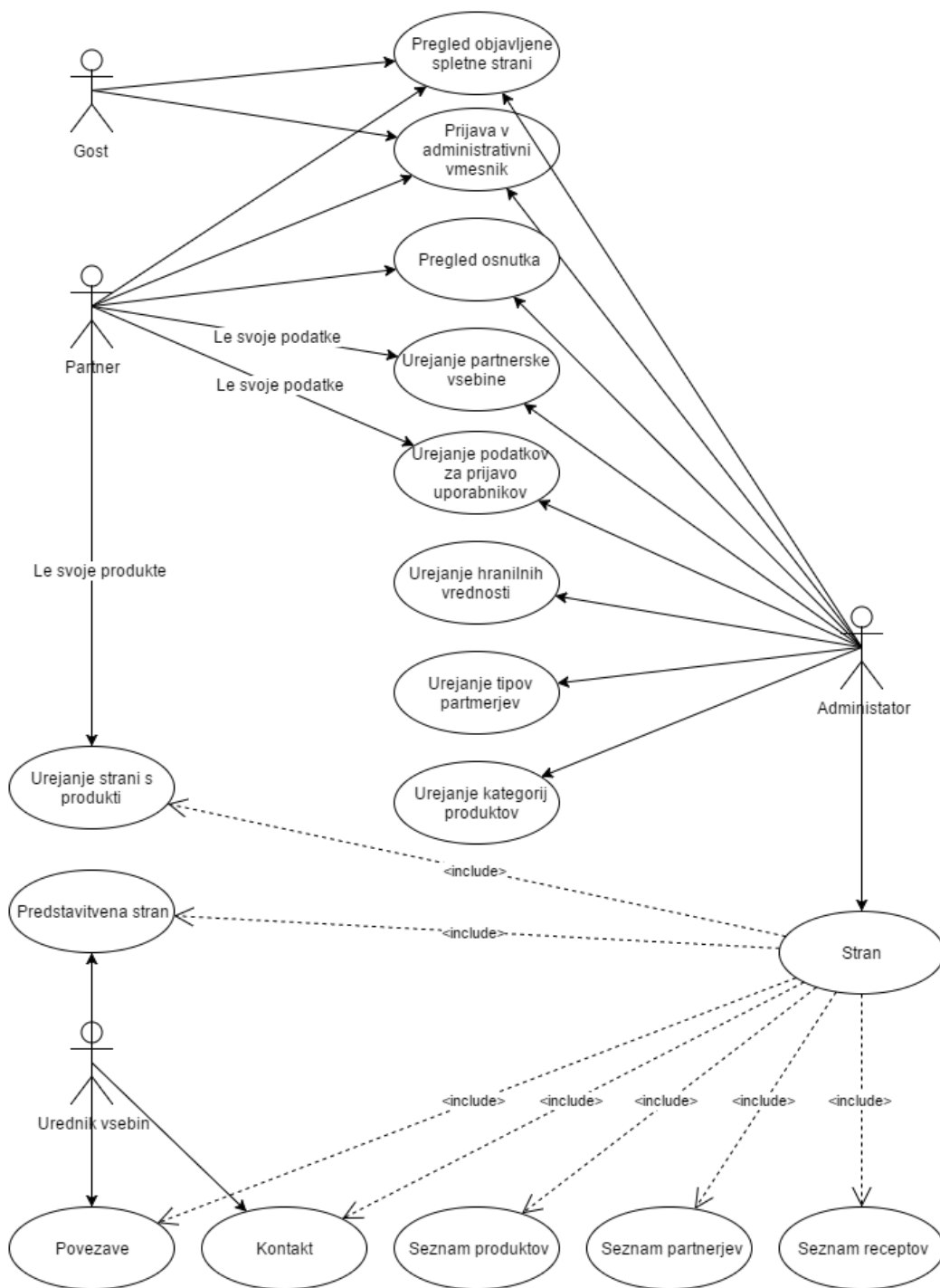
Uporabniki skupine partner lahko urejajo svojo vsebino, hkrati pa jim onemogočamo, da bi urejali druge, splošne strani na našem spletnem portalu.

Skupini smo dodelili pravice za urejanje vsebin le na svojem partnerskem objektu za urejanje partnerskih strani, ter na produktih, ki so povezani z njimi. Do drugih strani nimajo dostopa za urejanje, na voljo imajo le dostop za ogled.

Vsak partner, ki je predstavljen na strani, je hkrati lahko tudi uporabnik v administrativnem vmesniku, če mu dodelimo:

- elektronski naslov in geslo za dostop
- članstvo v skupini partner

Za potrebe urejanja portala, pa smo dodali še eno skupino, namenjeno urednikom področja akvakulture. Ti ne morejo posegati v podatke partnerjev, temveč lahko urejajo le splošno namenske strani, kot so “Predstavitvena stran”, “Stran za kontakt”, “Stran s povezavami”.



Slika 3.2: Primer uporabe sistema.

# Poglavje 4

## Implementacija

### 4.1 Podatkovni model

Spletni portal Akvakulture za delovanje uporablja programsko opremo Silver-Stripe. Ogrodje je zasnovano modularno, kar pomeni da lahko, ob pravilnem načrtovanju, izdelujemo in uporabljamo obstoječe module in si s tem poenostavimo in pohitrimo delo. Koda je objektno orientirana, kar je opazno tudi pri samem načinu programiranja v ogrodju. Osnovni paket programske opreme, za spletno stran je sestavljen iz:

- ogrodje (angl. *Framework*)
- modula za nastavitve (angl. *Site Config*)
- modula za razširjen administrativni del (angl. *CMS*)
- modula za poročila (angl. *Reports*)

V osnovi je celotna rešitev vezana na dedovanju razredov. Na nižje ležečih razredih so definirane operacije, ki višjim razredom omogočajo veliko avtomatizacije in s tem prihranijo čas, ter zmanjšajo možnost napak. V diplomski analogi se bomo omejili samo na višje razrede, kateri zadostijo skoraj vsem primerom uporabe. Vsak tak tip objekta lahko nosi različne podatke; ne-

kateri so podedovani, nekateri so definirani le na določenih objektih in ti se lahko zopet dedujejo.

Poleg takih lastnosti, so objekti med seboj lahko tudi v relacijah. Ker ima ogrodje razvito odlično orodje za preslikavo objektnih relacij, nam omogoča popolno avtomatično upravljanje s strukturo podatkovne baze. Če dodajamo oz. urejamo definicije podatkov na objektih, nam orodje zagotovi delovanje objekta in avtomatično ustvari nove tabele in stolpce v njih. To orodje nam omogoča tudi operacije iskanj po objektih na podoben način, kot da bi iskali po sami podatkovni bazi, s filtriranjem in sortiranjem rezultatov

Možnosti preslikav se odražajo glede na zmožnosti podatkovne baze, katera vsebuje vse dinamične podatke za spletno stran in njihove implementacije v SilverStripe. Na voljo imamo tako relacije ena-ena (angl. *has one*), ena-mnogo (angl. *has many*) ter mnogo-mnogo (angl. *many many*). Z njimi določimo odvisnost med objekti in kasneje to tudi uporabljamo.

Osnova za delovanje ogrodja SilverStripe je podatkovni objekt (angl. *DataObject*). Podatkovni objekt je tako osnoven tip objekta, da se sam brez dodatnih definicij ne zna prikazati na spletni strani (v MVC modelu manjkata krmilnik in pogled). Če želimo, lahko to logiko dodelamo, lahko pa njegove lastnosti izpisujemo posamezno, kot del predlog. Podatkovni objekt, je namenjen le temu, da vsebuje določene lastnosti za hrambo podatkov in je del modela v MVC strukturi.

Naslednji pomembnejši osnovni razred, kateri je razširjen iz podatkovnega objekta, pa je objekt tipa stran (angl. *Page*). To pa je objekt, kateri ima tudi svoj krmilnik in pogled, kar pomeni da je popolnoma samostojen za prikaz. Vsaka stran vsebuje pomembno informacijo, katera jo še ločuje od primarnega podatkovnega objekta – unikaten odsek spletnega naslova (angl. *URL Segment*).

Ko se torej odločamo za razvoj v ogrodju SilverStripe, nas samo ogrodje postavi pred odločitev, ali bomo za del potrebovali odsek spletnega naslova in bomo zato izbrali in razširili strani, ali pa bo to le del nečesa večjega in bomo izbrali in razširili podatkovni objekt.

Ogrodje nam omogoča tudi uporabo predlog, kar pomeni da je v praksi enostavno porazdeliti tako spletno stran med razvijalci ki skrbijo za programsko kodo, ter med razvijalci, ki skrbijo za izgled spletne strani. Tako kot pri pri programski kodi, se tudi tukaj uporablja podobna logika tudi pri predlogah. Te so lahko porazdeljene med moduli, vsak podedovan razred, pa bo za svoj izgled najprej uporabil predlogo ki se ujema z njegovim imenom, če pa ta ne obstaja, pa bo poiskal predhodne razrede iz podedovane strukture. V predlogah lahko tudi vključujemo druge predloge in s tem preprečimo večkratno pisanje enake kode.

Ko urejamo strukturo spletne strani z urejevalnikom vsebin SilverStripe, vidimo, da so strani porazdeljene v drevesni strukturi. Vsaka stran, razen če je določeno nasprotno s strani programerja, lahko leži neposredno na korenu spletnega mesta, ali pa je postavljena pod starševsko stran. Ker ima vsaka stran definiran svoj podatek za odsek spletnega naslova, tako gradimo spletni naslov za povezave na strani. Na primer: na domeni `www.primer.si` imamo na korenu spleta stran "O podjetju" s spletnim odsekom naslova "o-podjetju", pod to stran pa dodamo stran "Kontakt", s spletnim odsekom naslova "kontakt". Absolutni naslov strani z naslovom "Kontakt" bi bil `www.primer.si/o-podjetju/kontakt`.

## 4.2 Rešitev v SilverStripe

Spletni portal akvakulture v Sloveniji je grajen odzivno (angl. *Responsive*). To dosežemo s pametno postavitvijo elementov HTML in s predlogami CSS. S tem lahko omogočimo mobilnim napravam prijazno spletno stran - stran se lahko zoži na manjšo širino, pri tem pa poskrbi, da vseeno prikaže vse podatke. Prikazani podatki so enostavno berljivi, povezave imajo večje področje občutljivo na klik oz. dotik (angl. *Touch*). Velika prednost tega načina izdelave spletne strani je v tem, da sta v ozadju ista HTML ter PHP koda za vse vrste naprav, prav tako pa tudi ni podvajanja podatkov v podatkovni bazi. Le v kaskadnih stilskih predlogah so zapisi za oblikovanje definirani za

različne tipe naprav. Stili se pravzaprav vklaplajo in izklaplajo glede na širino naprave ali pa širino prikazanega prostora v brskalniku, kar dosežemo z medijskimi poizvedbami (angl. *Media query*) vgrajenimi v CSS3.

SilverStripe nudi le ustrezne programske komponente, ne pa tudi celovite rešitve, zato smo morali razviti celoten podatkovni model in s tem tudi primerne razrede, kateri predstavljajo tako objekte, kot tudi strani. Ob razvoju je SilverStripe sam poskbel za ustrezno izgradnjo tabel v podatkovni bazi.

Tako smo morali za potrebe prevajanja, razviti poseben razred '*Decorator*', kateri nam ponudi enake operacije nad več objekti, ne da bi s tem posegali direktno v sam objekt. Okrasni razred sam zazna polja namenjena prevajanju, jih doda v bazo in jih doda k izpisu v administrativnem vmesniku. Ta razred smo dodali na podatkovne objekte po potrebi.

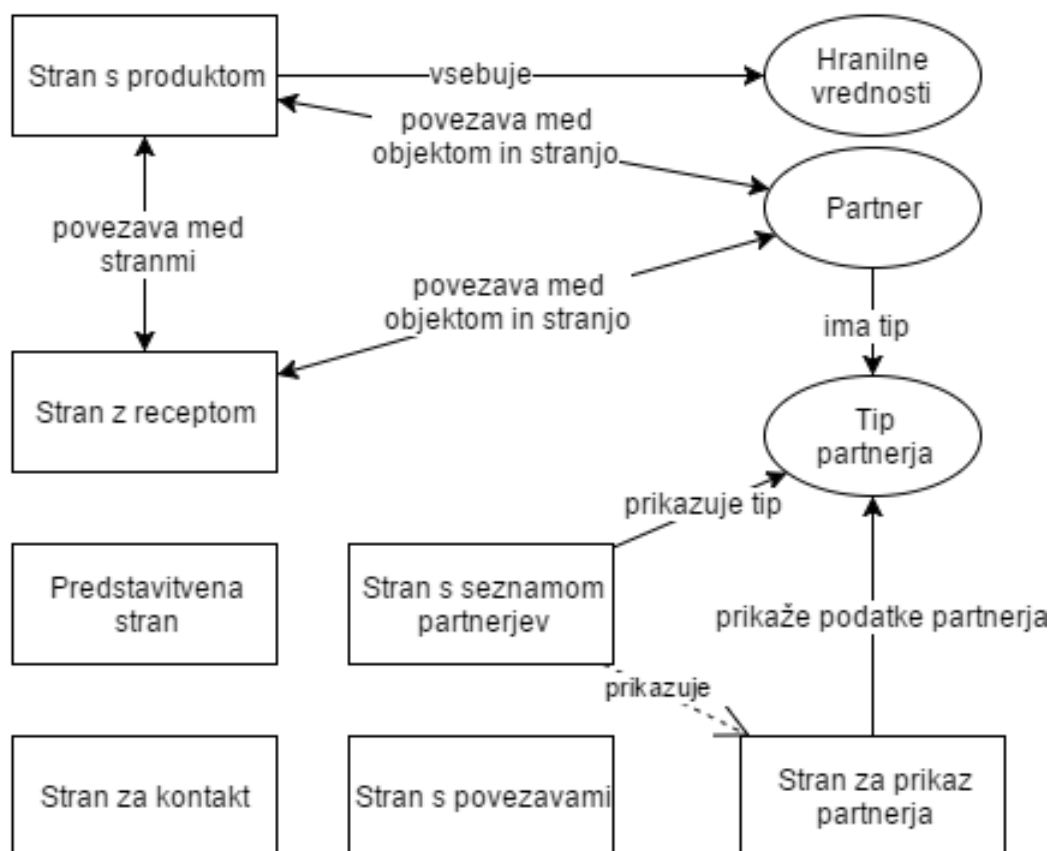
Določili smo ustrezne razrede za:

- strani: če se bo posamezen objekt razreda prikazoval samostojno in le na eni strani v vsaki jezikovni veji;
- podatkovne objekte: vsi drugi, kateri ne ustrezajo zgornjemu pogoju.

Za potrebe predstavitvenih strani (slika 4.1), smo razvili stran tipa "Predstavitvena stran", katera poleg naslova vsebuje še besedilo in možnost vstavljanja slik. Za prikaz partnerjev, smo potrebovali stran, ki bo sposobna prikazati vse informacije ki jih želimo. Stran tipa "Seznam partnerjev" je sistemska in služi v drevesni strukturi zato, da dobimo svoj spletni segment naslova, na katerem prikazujemo partnerje. Stran s seznamom partnerjev se preko objekta "Tip partnerja", poveže z objekti partnerjev, ter tako dobi vsebino za prikaz na strani. Stran za seznam tako nima direktne vsebine za urejanje, razen naslova, ter nastavitev prikaza zemljevida (x in y koordinati sredine zemljevida, začetna oddaljenost) vseh partnerjev.

Objekt "Tip partnerja", tako služi kot povezava, med stranjo in podatkovnim objektom tipa "Partner". Objektu smo dodali le podatek o naslovu, preko katerega lažje urejamo te povezave v administrativnem vmesniku. Na drugi strani povezave, pa imamo, kot že omenjeno objekt tipa "Partner", ka-





Slika 4.1: Medsebojna odvisnost strani in objektov.

teri nosi vse informacije partnerja za prikaz na slovenski in angleški različici strani. Poleg nekaterih skupnih podatkov, kot so naslov, logotip (slika), izpostavljen produkt, ima objekt tudi polja za prevode. Za prevedeni polji, smo določili polje za vnos telefonske številke in polje za predstavitev podjetja. Poleg teh polj, smo razredu dodali še povezavi z razredoma "Lokacija" (angl. *Location*), kateri nam predstavlja točko na zemljevidu, ter "Zavihek" (angl. *Partner Tab*), s katerim lahko partner sam doda poljubne vsebinske zavihke k svoji predstavitvi. Tudi ti zavihki so prevedeni v oba jezika, kar pomeni da lahko vsebino zavihka uredi ločeno za slovenski in ločeno za angleški jezik. Tem zavihkom lahko spreminjamo vrstni red in s tem določamo kateri bo prvi in kateri zadnji. Poleg naštetega, pa ima vsak partner lahko tudi svoje

produkte in recepte, kar v našem primeru predstavlja povezavo s stranmi produktov in receptov; to so strani tipa ‘produkt’ (angl. *Product Page*) in ‘recept’ (angl. *Recipe Page*).

Strani za recept, so v naši drevesni strukturi pod stranjo s seznamom receptov, to je stran “Seznam receptov”, strani s produkti pa pod seznamom produktov, to je stran tipa “Seznam produktov”. Vsebina obeh strani s seznamom je zgrajena avtomatično, zato uredniško razen naslova nimata nastavitev. Vsebina pri teh dveh tipih strani, se črpa iz podstrani, kar na uporabniški strani ustvari mrežo produktov.

Posamezna stran za recept ima nekoliko več polj za vnos podatkov, da lahko na strani primerno prikažemo vso vsebino. Strani smo dodali povezavo s partnerjem, polje za vnos časa priprave jedi, polje za izbiro zahtevnosti jedi, sliko za seznam, ter veliko vnosno polje za vnos postopka priprave jedi. Poleg naštetega, na recept lahko pripnemo produkte (druge strani) in tako ustvarimo povezavo med produktom in receptom.

Tako kot na receptih, lahko drugo stran te povezave urejamo tudi na produktu. Za primer si vzemimo stran tipa “Ribja stran”, s katero vnašamo produkte akvakulture. Stran za izgradnjo seznama na vključenih straneh potrebuje kratek opis, naslov in sliko. Ločeno, daljšo vsebino lahko pripravimo v ločenem polju, katera se izpiše na strani produkta. Hkrati smo želeli poleg vsebine izpisati še druge podatke produkta, kot so identifikator hrane, kategorijo hrane, ter hranilne vrednosti. Ker so hranilne vrednosti potrebne pri vseh produktih, so realizirani s podatkovnimi objekti, kateri podobno kot objekt tipa partner, vsebujejo polje za prevod.

Posamezen objekt hranilnih vrednosti vsebuje polje za angleščino, polje za slovenščino, ter za potrebe sinhronizacije z OPKP naziv, kateri se ujema z njihovim izpisom. Vsakemu posameznemu objektu, lahko tudi določimo ali bo na strani skrit ali ne. Objekt sam po sebi ne vsebuje podatka o količini hranilne vrednosti, ampak je ta podatek zapisan v povezovalni tabeli hranilne vrednosti s produktom. Povezava s produktom je namreč mnogo-mnogo, kar v podatkovni bazi ustvari dodatno tabelo, kjer se ta dva objekta povežeta.

## Poglavje 5

# Portal slovenske akvakulture

Spletni portal slovenske akvakulture je sestavljen iz dveh ločenih delov, ki sta med seboj povezana z isto podatkovno bazo. To sta administrativni vmesnik in uporabniški spletni portal. Slednji je namenjen širši javnosti za predstavitev področja akvakulture v Sloveniji in njenih ponudnikov, medtem ko je administrativni vmesnik namenjen le uporabnikom, kateri imajo pravice za dostop in lahko urejajo vsebino, ki je prikazana na uporabniški strani.

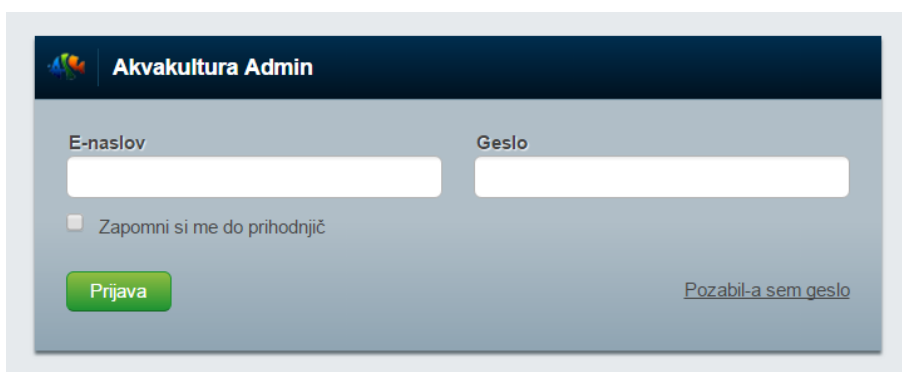
### 5.1 Administrativni vmesnik

Administrativni vmesnik je ločena spletna stran, namenjena le avtenticiranim uporabnikom, kateri imajo omogočen dostop. To je poseben del spletnega portala, ki je namenjen urejanju vsebin, katere se prikazujejo v uporabniškem delu spletnega portala. Pred uporabo, se mora uporabnik prijaviti, nato pa lahko prične z urejanjem vsebine, za katere ima dodeljene pravice.

#### Prijava v administrativni vmesnik

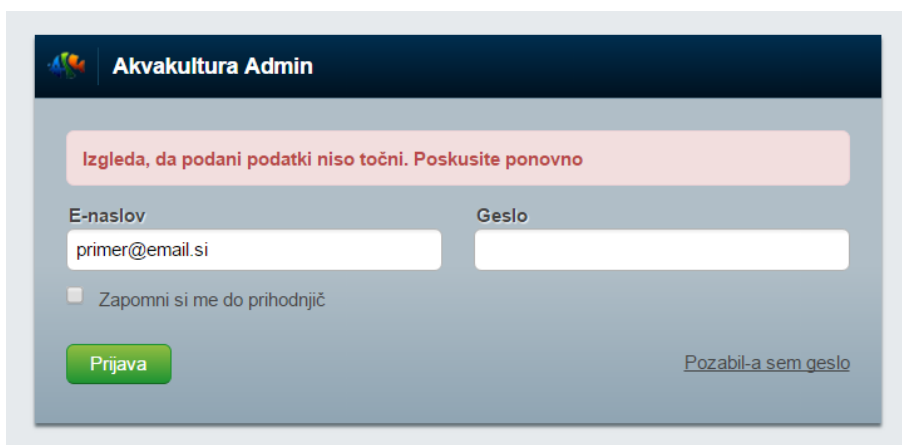
Do vmesnika lahko dostopamo na domačem naslovu spletne strani, z dodatkom URL parametra */admin*.

Na zaslonu je okence (Slika 5.1), katero zahteva vnos elektronskega naslova, ter gesla. Dostopamo do dela strani, kateri je namenjen le upravljalcem

The image shows a login form for 'Akvakultura Admin'. At the top, there is a dark blue header with a logo on the left and the text 'Akvakultura Admin' on the right. Below the header, the form has two input fields: 'E-naslov' (Email) and 'Geslo' (Password). Below the 'E-naslov' field, there is a checkbox labeled 'Zapomni si me do prihodnjič' (Remember me until next time). At the bottom left of the form is a green button labeled 'Prijava' (Login). At the bottom right is a link labeled 'Pozabil-a sem geslo' (Forgot my password).

Slika 5.1: Prijavno okno. Poziv k vpisu elektronskega naslova ter gesla.

spletnega portala. Pred vpisom lahko obkljukamo možnost “Zapomni si me do prihodnjič”, kar nam nastavi piškotek v brskalniku, za čas treh mesecev. Kadarkoli bomo v času teh treh mesecev od prijave odprli administrativni vmesnik, nas bo sistem avtomatično prijavil. Po vpisu podatkov, kliknemo na gumb “Prijava”, kateri sproži preverjanje vpisanih podatkov in postopek prijave.

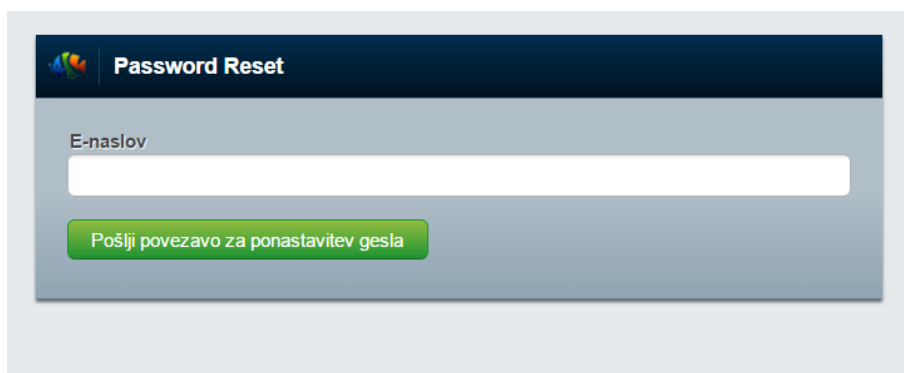
The image shows the same login form as in Slika 5.1, but with an error message displayed at the top. The error message is in a pink box and reads: 'Izgleda, da podani podatki niso točni. Poskusite ponovno' (It looks like the provided data is not correct. Please try again). The 'E-naslov' field now contains the text 'primer@email.si'. The 'Geslo' field is empty. The 'Prijava' button and the 'Pozabil-a sem geslo' link are still present at the bottom.

Slika 5.2: Neuspešna prijava, izpis obvestila z obrazložitvijo, zakaj je prišlo do napake.

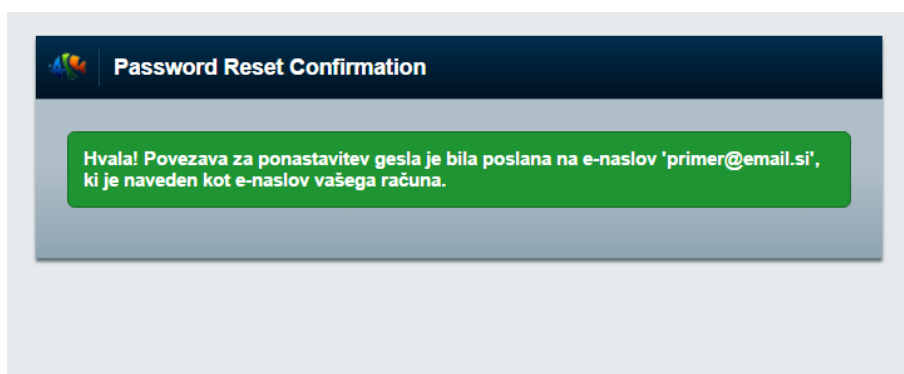
Če je prijava neuspešna, nas preusmeri nazaj na prijavno okno (Slika 5.2), z opisom napake zakaj prijava ni bila uspešna. Lahko poskusimo ponovno

s prijavo, ali uporabimo obrazec za pozabljeno geslo. Pogosti razlogi, zakaj prijava ni uspešna so:

- uporabnik ne obstaja,
- geslo ni pravilno
- uporabnik nima dovoljenja za dostop



Slika 5.3: Obrazec za pozabljeno geslo.



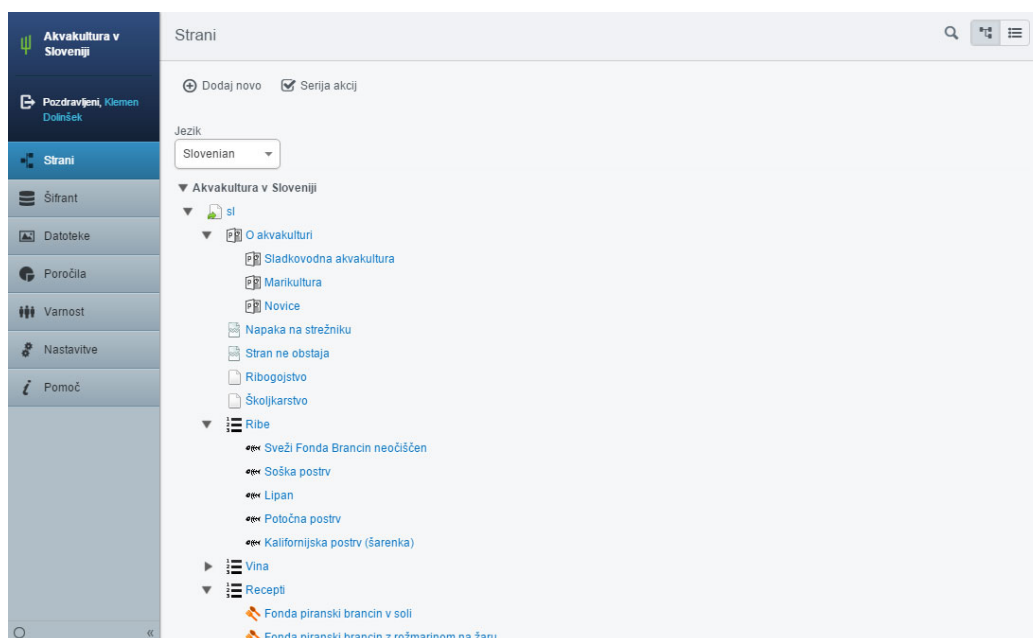
Slika 5.4: Obvestilo o rezultatu pošiljanja elektronskega sporočila.

V kolikor smo pozabili geslo za naš dostop, lahko uporabimo povezavo “Pozabil-a sem geslo” (Slika 5.3). Prikaže se nam drugo okence, s pozivom

k vpisu našega elektronskega naslova. Po vpisu kliknemo gumb "Pošlji povezavo za ponastavitev gesla". Po končanem postopku, nam sistem prikaže okence z obvestilom o uspešni prijavi (Slika 5.4).

Če je prijava neuspešna, nas preusmeri nazaj na prijavno okno (Slika 5.2), z opisom napake zakaj prijava ni bila uspešna. Lahko poskusimo ponovno s prijavo.

## Pregled administrativnega vmesnika



Slika 5.5: Osnovni pogled administrativnega vmesnika kot ga vidi uporabnik skupine administrator.

Vmesnik je v osnovi razdeljen na dva dela (Slika 5.5):

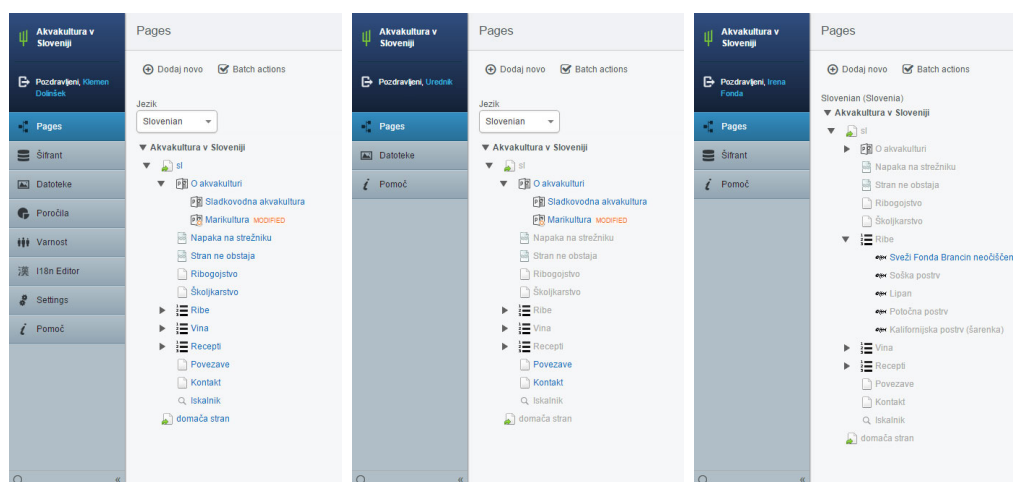
- navigacija v skrajno levem modrem delu
- vsebina v desnem delu vmesnika

Navigacija je namenjena preklopu med posameznimi pogledi. Med delom, jo lahko pomanjšamo in tako pridobimo dodaten prostor za urejanje vsebine

v desnem delu. Navigacija za menjavo pogledov, ponuja več možnosti, a se bomo za potrebe diplomske naloge omejili le na “Strani”, “Šifrant” , “Varnost” in “Nastavitve”.

Vsebinski del se spreminja glede na izbrani pogled. Prikaže lahko seznime, posamezne objekte, ali kombinacijo obeh. Če izbrani pogled to omogoča, je vsebina porazdeljena po zavihkih (skrajno desni zgornji rob), lahko pa vsebuje tudi dvonivojsko porazdelitev po zavihkih.

Glede na uporabniško skupino kateri pripadamo kot prijavljen uporabnik, imamo lahko omejen dostop do nekaterih sklopov. V navigaciji in vsebinskem delu se lahko določeni odseki skrijejo, če skupina ne omogoča dostopa (Slika 5.6).



(a) Administrator

(b) Urednik vsebin

(c) Partner

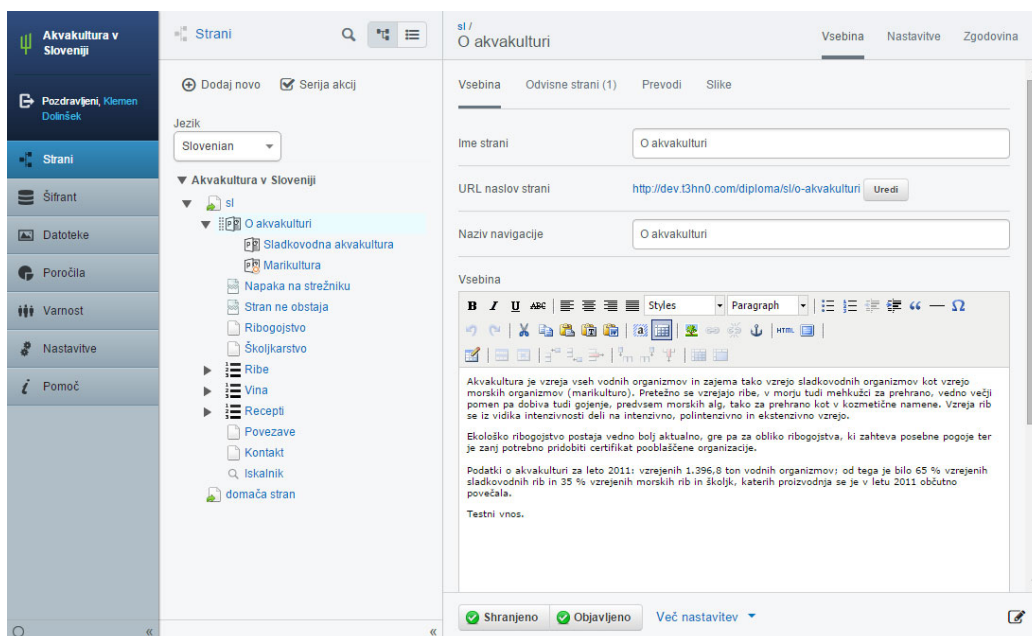
Slika 5.6: Pogled, kot ga vidi član določene skupine.

## “Strani”

“Strani” omogočajo urejanje strani in vsebine spletnega portala (Slika 5.7). Vsebinski del je razdeljen na dva dela:

- levi del s filtrom, menjavo pogleda, menijem za izbiro jezika ter drevesno strukturo vseh strani

- desni del prikazuje vsebino in nastavitve trenutno izbrane strani za urejanje



Slika 5.7: Pogled in urejanje strani “O akvakulturi”.

V levem delu je s klikom na gumb “Dodaj novo” mogoče dodati novo stran. Dodajanje nove strani se odpre v obliki seznama (Slika 5.8), kjer lahko izberemo željen tip, katerega bi ga radi dodali. Določene strani imajo onemogočen posamezen tip, ker ga tam ni smiselno uporabiti (tip “Produkt” na strani o “O akvakulturi” ni na voljo).

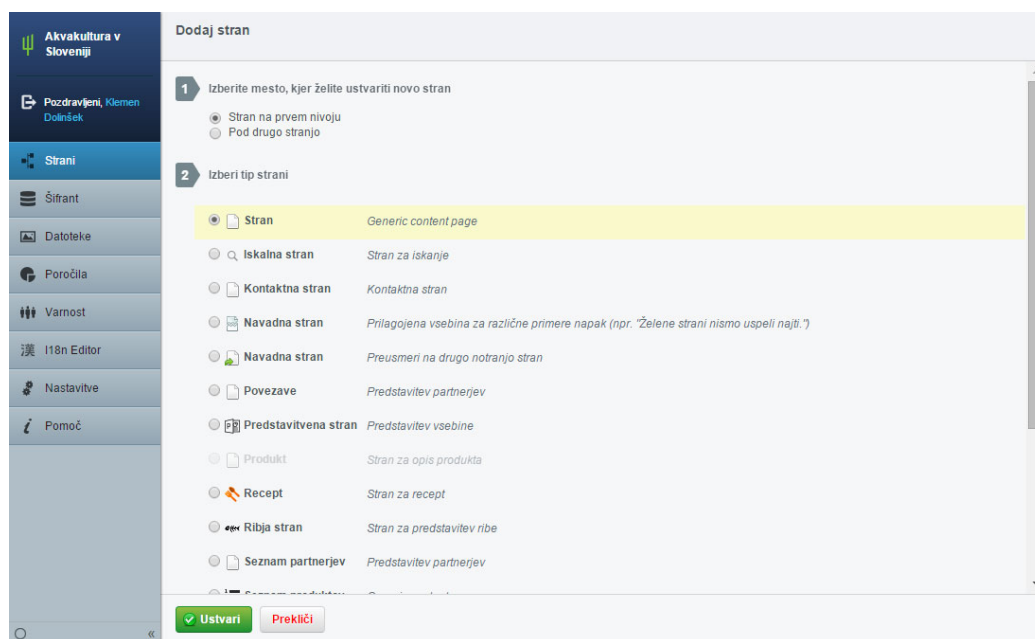
Po izboru tipa, kliknemo na zelen gumb “Ustvari”, kateri vodi na drugi in hkrati zadnji korak, urejanje podatkov za prikaz na strani. Ti podatki so odvisni glede na izbran tip strani. Za primer dodajanja, si bomo pogledali “Ribjo stran” (Slika 5.9). Dodali smo jo pod stran z naslovom “Ribe”.

Lahko pa z desnim klikom na željeno starševsko stran preskočimo en korak in v meniju, ki se odpre izberemo točno določen tip strani.

Vsaka stran ima tri stopnje:

- osnutek: stran je vidna v drevesni strukturi, na strani je skrita

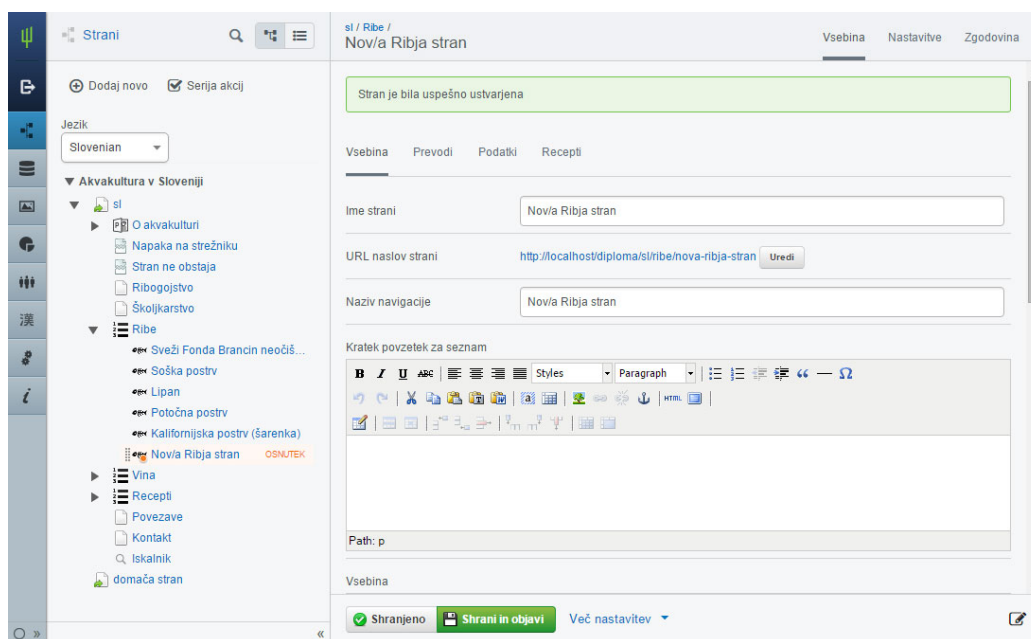




Slika 5.8: Seznam podstrani, katere lahko dodamo na trenutno mesto. Izbrali smo starševsko stran “sl”, kjer je tip “Produkt” onemogočen.

- objavljena stran: stran je vidna v drevesni strukturi in tudi na strani
- arhivirana stran: stran ni vidna v drevesni strukturi, na strani je skrita

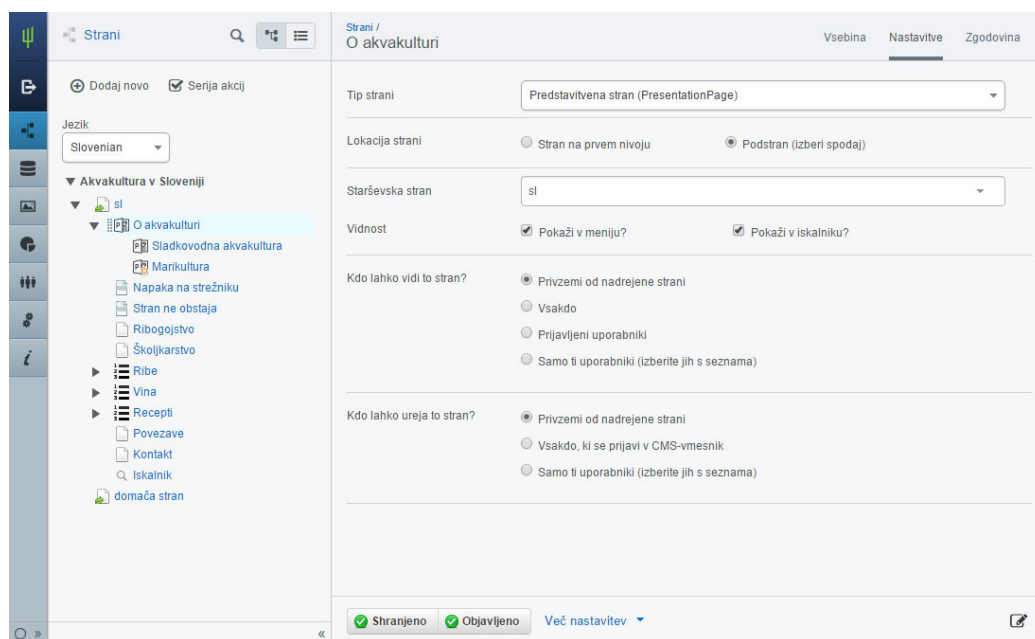
Ko dodamo novo stran, ta dobi status osnutek. V tej fazi, lahko kot administrator kliknemo na povezavo pri opisu “URL segment strani”, kar nam odpre predogled in s tem možnost da preverimo kako se obnaša vsebina na strani. Ko smo zadovoljni z izgledom, lahko kliknemo na gumb “Shrani in objavi”, kar bo objavilo stran s trenutnimi vsebinami in jo tako naredilo dostopno na spletu. Če strani ne potrebujemo več, jo lahko začasno umaknemo iz objave, s klikom na “Več nastavitev”, ter “Umakni iz objave”. Če strani res ne potrebujemo več in bi jo radi izbrisali, to naredimo s klikom na “Več nastavitev”, ter “Arhiviraj”. Stran se bo skrila tako iz spletnega mesta (v kolikor je bila objavljena) ter iz drevesne strukture. Taka stran je še vedno lahko dostopna vsem uporabnikom administrativnega, če filtrirajo podatke drevesnega seznama s klikom na lupo, nad drevesnim seznamom.



Slika 5.9: Primer dodajanja “Ribje strani”. Prikazana so polja, namenjena za prikaz te strani.

Kot administrator, imamo dostop do vseh podstrani, kjer lahko urejamo njihove podatke, če pa smo prijavljeni kot zunanji partner, pa smo omejeni le na urejanje določenih tipov podstrani. Obstajajo strani različnega tipa, kjer ima lahko vsak tip svoje nastavitve in podatkovno strukturo, ter svojo predlogo za prikaz na strani. Ena izmed obveznih strani, ki je potrebna za delovanje portala, je stran, katera je določena z url segmentom “home”. V našem primeru take strani ne potrebujemo, zato smo za stran “home” izbrali preusmeritveno stran, katera nas preusmeri na dejansko prvo stran – O akvakulturi.

Vsaka stran ima tudi svoje nastavitve (Slika 5.10), kjer lahko urejamo kdo vidi to stran, kdo jo lahko ureja, nadzorujemo ali se stran kaže v meniju, lahko pa tudi spremenimo tip strani.

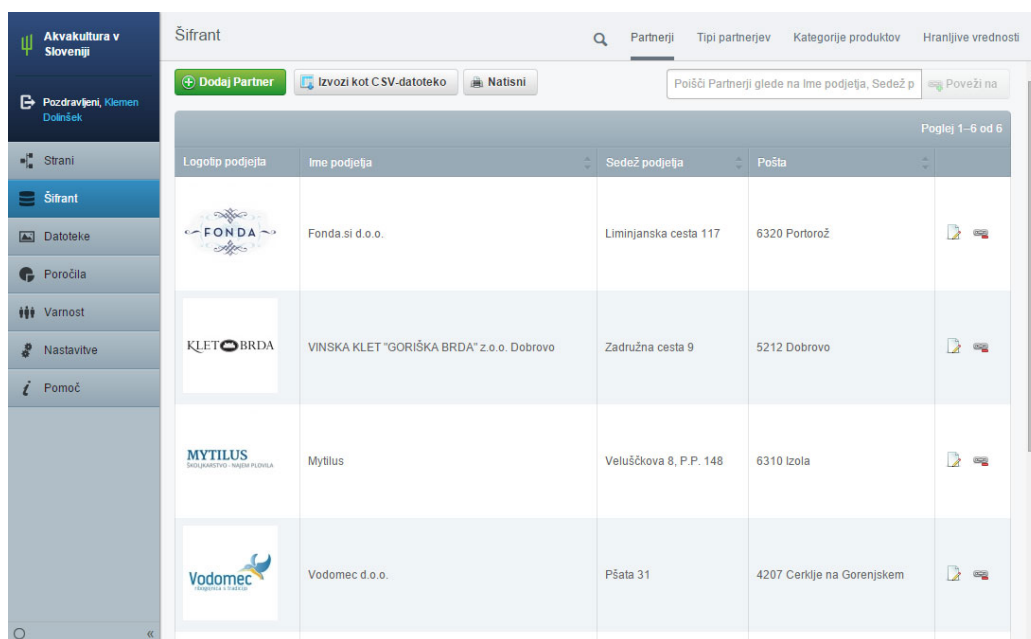


Slika 5.10: Urejanje nastavitev izbrane podstrani.

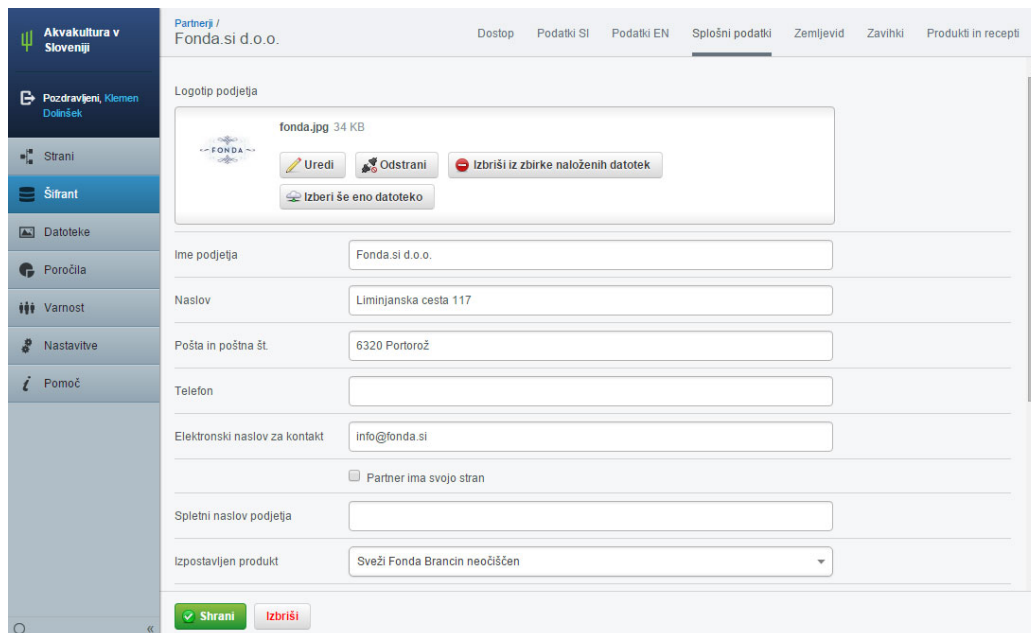
### “Šifrant”

Do tega dela vmesnika imajo popoln dostop le administratorji, partnerji pa imajo omejen pogled. Vidiijo in urejajo lahko le svoj vnos. Tu lahko urejamo obstoječe in dodajamo nove partnerje, kategorije produktov in hranilne vrednosti (Slika 5.11). Šifrant služi kot skladišče pogosto uporabljenih objektov, katere lahko uporablja več objektov in strani, ali pa za bolj enostaven pregled in filtriranje objektov večjega števila objektov istega tipa.

Desni del tega pogleda, je zopet ločen na dva dela; zgornji del, namenjen filtriranju vnosov ter izboru vrste objektov in spodnji s seznamom vnosov. Ko dodajamo oz. urejamo obstoječ vnos (Slika 5.12), zgornji del izgine oz. se zamenja z delom, ki je določen za izbran vnos. Po dodajanju partnerja, mu lahko v pogledu varnost (Slika 5.13) določimo še geslo, ter pravice in mu tako omogočimo dostop do administrativnega vmesnika, da lahko sam ureja svoje podatke in produkte.

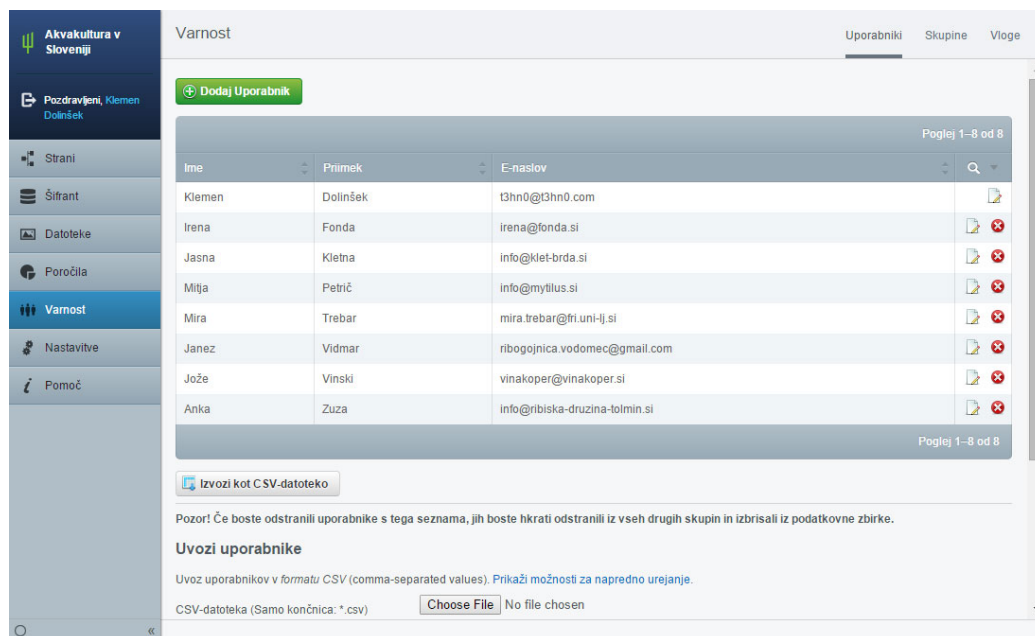


Slika 5.11: Šifrant, osnovni pregled.



Slika 5.12: Urejanje partnerja. Na sliki vidimo zavihke za različne sklope vsebin, v vsakem zavihku pa so različne nastavitve.

## “Varnost”



Slika 5.13: Pogled varnost: seznam uporabnikov, kateri imajo račun v administrativnem vmesniku.

Pogled varnost (Slika 5.13) je namenjen upravljanju uporabnikov, skupin in vlog, ter urejanju dostopov do administrativnega vmesnika. Vsakemu uporabniku, za katerega imamo pravice, lahko spreminjamo dostopne podatke (Slika 5.14), ter urejamo jezik vmesnika in nastavitve prikaza datuma ter časa. Na sliki vidimo izbrani jezik vmesnika, skupino za dostop do administrativnega vmesnika, ter izbran format časa in ure.

## “Nastavitve”

Pogled je namenjen urejanju osnovnih podatkov (Slika 5.15), kateri se nanašajo na celotno stran. Med take podatke štejemo ime strani, katero v našem primeru prikazujemo v glavi spletnega portala, ter nad orodno vrstico v brskalniku. Tu lahko tudi določamo splošen dostop do strani, ali je dostopna le prijavljenim uporabnikom, ali pa vsem, kar je privzeta nastavitve. Tu je

The screenshot displays the 'Varnost / Dolinšek, Klemen' settings page. The left sidebar contains navigation links: 'Akvakultura v Sloveniji', 'Pozdravljeni, Klemen Dolinšek', 'Strani', 'Šifrant', 'Datoteke', 'Poročila', 'Varnost' (highlighted), 'Nastavitve', and 'Pomoč'. The main content area includes the following fields and options:

- Ime:** Text input with 'Klemen'.
- Preimek:** Text input with 'Dolinšek'.
- E-naslov:** Text input with 't3hn0@t3hn0.com'.
- Spremeni geslo:** Section header.
- Datum zadnjega obiska:** Displayed as 'feb 29, 2016 9:21:27 pop.'.
- Jezik:** Dropdown menu set to 'Slovenian (Slovenia)'.
- Failed Login Count:** Text input with '0'.
- Skupine:** Text input with 'Administrators x'.
- Format datuma:** Radio buttons for 'Feb 29, 2016', '2016/02/29', and '02/29/2016'. A selected option 'Prilagojeno' (Custom) is shown with a format 'MMM d, y' and a note '(Predogled: "Feb 29, 2016")'. A link 'Show formatting help' is present.
- Format časovnega zapisa:** Radio buttons for '9:21 PM', '21:21', and '21:21:28 (privzeto)'. A selected option 'Prilagojeno' (Custom) is shown with a format 'h:mm:ss a'.
- Shrani:** A green button with a checkmark to save the changes.

Slika 5.14: Urejanje uporabnika Klemen Dolinšek.

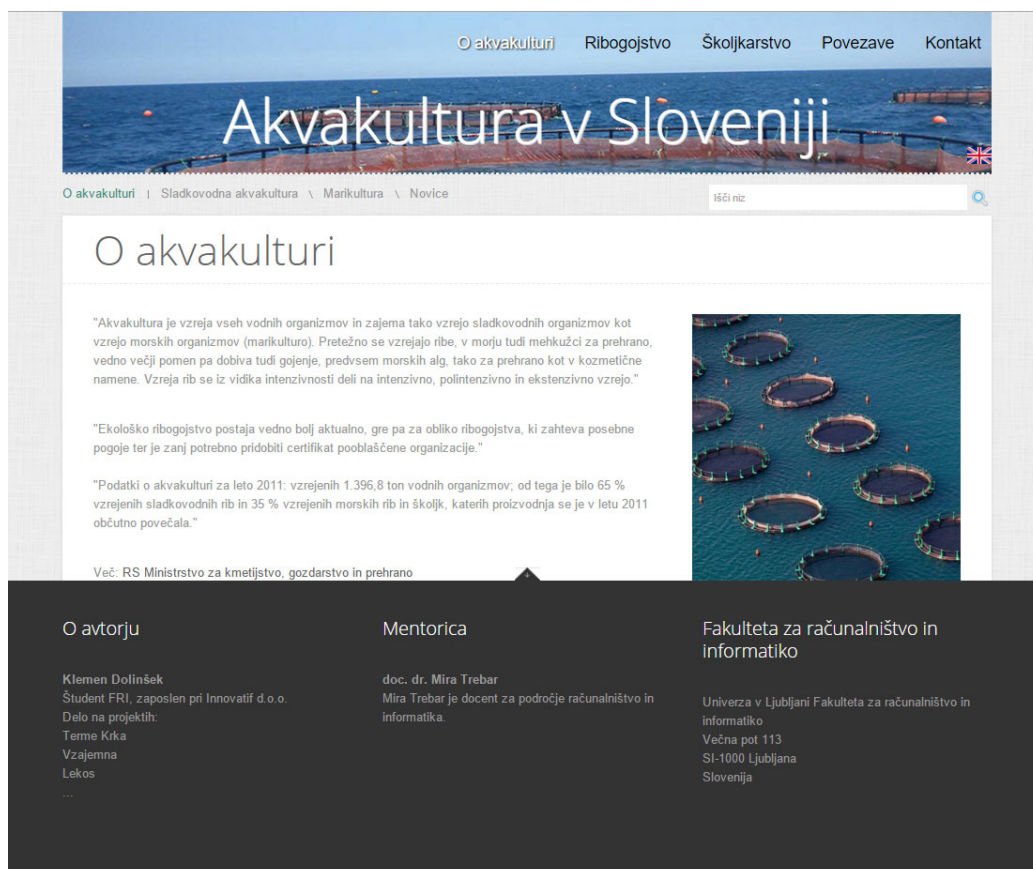
dodana tudi možnost spreminjanja ključa za storitve Google zemljevidi, ter možnost prevažanja nastavitev v druge jezike. Vsaka jezikovna veja ima namreč svoje nastavitve naslova, dostopnosti ter drugih prej opisanih možnosti.

The screenshot displays the 'Nastavitve' (Settings) page of the 'Akvakultura v Sloveniji' administrative interface. The interface has a dark blue sidebar on the left with the following menu items: 'Strani', 'Šifrant', 'Datoteke', 'Poročila', 'Varnost', 'Nastavitve' (highlighted), and 'Pomoč'. The main content area is titled 'Nastavitve' and includes a top navigation bar with links: 'Domov', 'Dostop', 'Google', and 'Prevodi'. The settings form contains three fields: 'Naslov spletnega mesta (title)' with the value 'Akvakultura v Sloveniji', 'Slogan' which is empty, and 'Tema' with a dropdown menu showing '(Uporabi privzeto temo)'. A green 'Shrani' button with a checkmark is located at the bottom of the form.

Slika 5.15: Nastavitve: urejanje splošne nastavitvev strani.

## 5.2 Uporabniški dostop

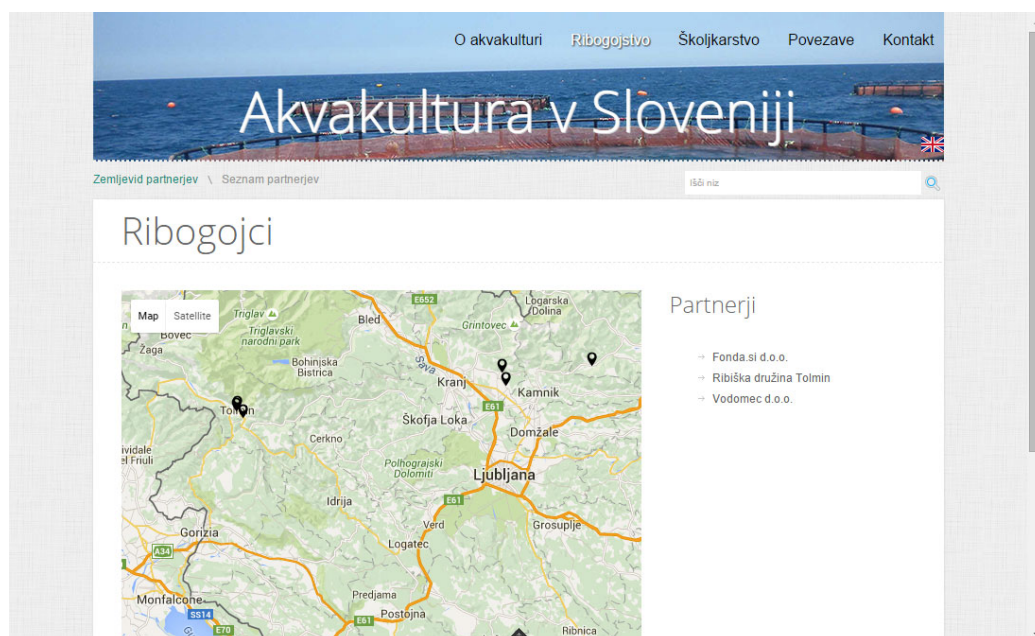
Javna aplikacija (Slika 5.16) je dostopna vsem uporabnikom in je namenjena za prikaz informacij, ponudnikov in njihovih storitev ter produktov. Razdeljena je na tri glavne dele: glava, vsebina in noga. V glavi je prikazana glavna navigacija, s katero se lahko premikamo med rubrikami. Glede na izbrano rubriko, se nam bo prikazala ustrezna predloga in vsebina. Glede na tip predloge, delimo stran na sedem različnih strani na tem nivoju. Najbolj osnovna stran, je stran tipa “predstavitvena stran” (Slika 5.16), katera nam omogoča izpis enostavnega besedila, z možnostjo dodajanja slik.



Slika 5.16: Predstavitvena spletna stran z opisi področja akvakultura in sliko ribogojnice.

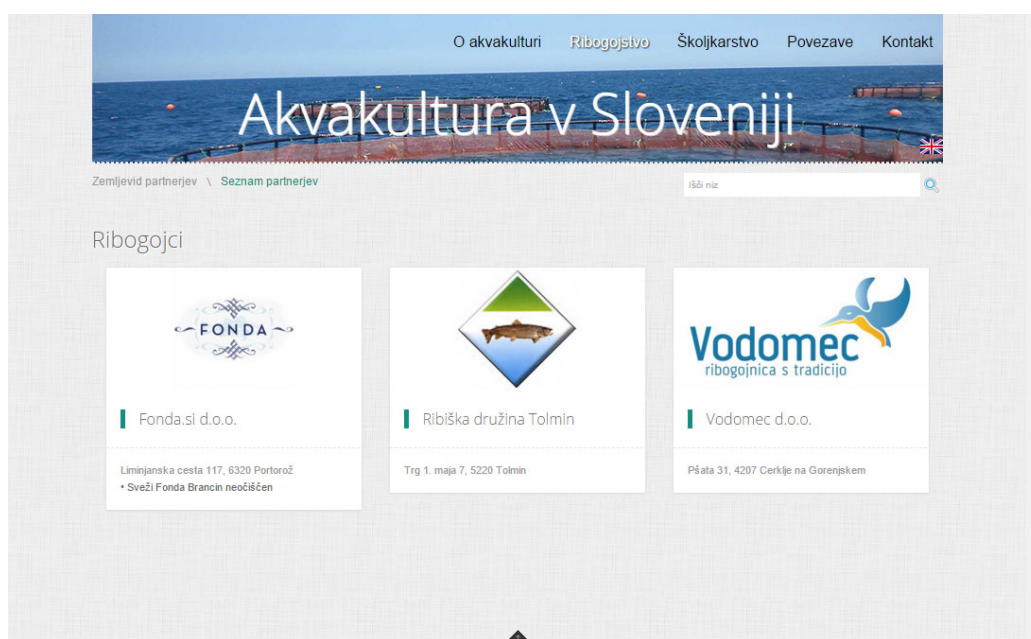


Naslednji tip strani, je stran s seznamom partnerjev. Odvisen je od nastavitve na strani, katere kategorije prikazuje. Na voljo sta dve tovrstni strani, ena s prikazom ribogojcev in drugo s prikazom školjkarjev. Za primer si bomo ogledali stran ribogojcev (Slika 5.17), a delovanje obeh je enako, le vsebina je drugačna. Stran s seznamom nam privzeto odpre stran z zemljevidom ter s seznamom ribogojcev ob zemljevidu. Na zemljevidu so označene dodane lokacije ribogojcev s seznama. S klikom na označbo na zemljevidu, se nam v desnem seznamu obarva izbran partner in nam tako olajša iskanje partnerjev preko zemljevida.



Slika 5.17: Seznam ribogojcev na zemljevidu.

Stran s seznamom partnerjev pa ima še en način predstavitve partnerjev, do katerega pridemo s klikom na povezavo "Seznam partnerjev". Odpre se nam mrežni seznam partnerjev (Slika 5.18), zopet le tistih, kateri so v kategorijah prikazanih na tej strani. Mrežni seznam nam ponuja možnost izpisa več podatkov partnerja, med drugimi tudi logotip, ter do en izpostavljen produkt.



Slika 5.18: Seznam ribogojcev.

Ob kliku na partnerja iz prvega ali drugega pogleda, nas stran preusmeri na njegovo stran, ki je bodisi na našem portalu, ali pa je to povezava, katera na pelje na njegovo, ločeno spletno stran. Ker je portal predvsem namenjen manjšim podjetjem, ki so še brez svoje spletne strani, jim na tem portalu lahko omogočimo njihovo samostojno predstavitevno stran. V kolikor pa podjetje že ima svojo spletno stran, pa obiskovalca preusmerimo tja.

Vsak partner, ki se tako odloči, da bi imel predstavitevno spletno stran na našem portalu, ima na voljo nadzor nad svojimi vsebinami za prikaz, na voljo pa ima tudi svoj spletni naslov, ki je sicer unikatni del portala akvakultura v Sloveniji. Testni primer s predstavitevno stranjo na našem portalu za predstavitev diplomskega dela, predstavlja partner Fonda d.o.o.

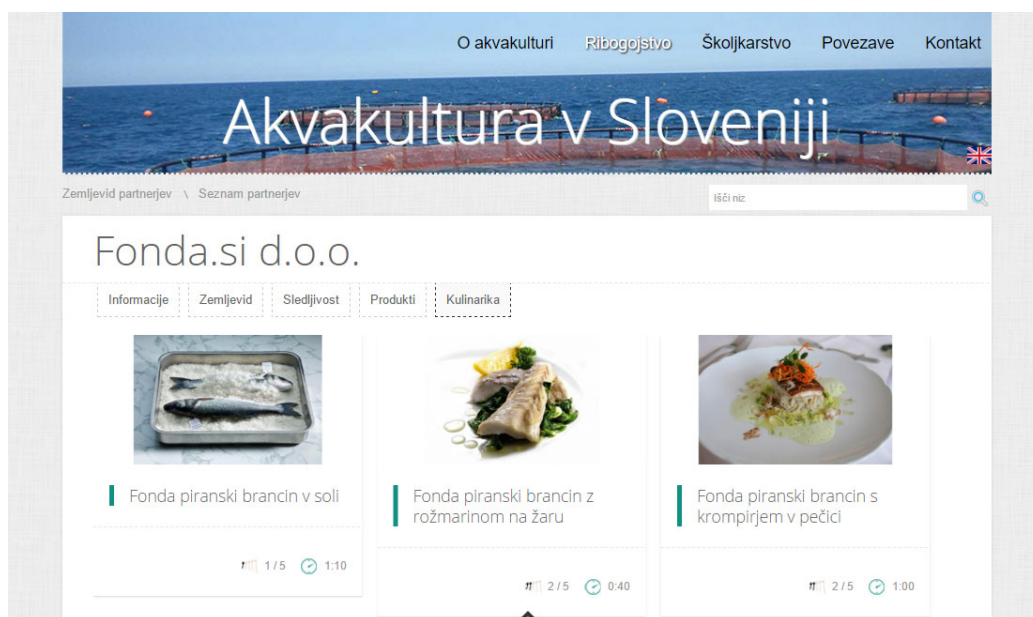
Podstran vsakega partnerja (Slika 5.19), je sestavljena iz zavihkov, kjer vsak zavihek vsebuje ločen del podatkov. Kot prvi, se nam odpre zavihek "Informacije", ker se izpišejo podatki o podjetju, kot pa tudi osnovno besedilo za opis podjetja. Naslednji zavihek, je zavihek "Zemljevid", kjer so označene



Slika 5.19: Primer spletne strani podjetja Fonda.si d.o.o.

vse dodane lokacije za izbranega partnerja. V kolikor partner nima dodanih lokacij, se zavihek zemljevid ne prikaže. Oba omenjena zavihka sta sistemska, kar pomeni, da ju portal doda sam, v kolikor so izpolnjeni določeni podatki. Za zemljevidom, pa je možnost, da si uredniki sami dodajajo vsebinske zavihke. V našem primeru, smo dodali zavihek z nazivom “Sledljivost”, kamor smo vnesli besedilo, ki se nanaša na to temo. Stran je lahko brez vsebinskih zavihkov, ali pa jih je več, odvisno od števila vnosov. Na mestu za uporabniško dodanimi zavihki, pa sta zopet dva sistemska zavihka, “Produkti” in “Kulinarika” (Slika 5.20).

Stran za predstavitev produkta (Slika 5.21), je podobno, kot predstavitvena stran za partnerja, porazdeljena po zavihkih. Zavihek informacije nam prikaže veliko sliko produkta, osnovne podatke gojitelja, ter opis izdelka. Naslednji zavihek, nam prikazuje podatke o hranilnih vrednostih živil. V našem primeru, smo hranilne vrednosti pridobili iz baze spletne strani “Odprte platforme za klinično prehrano” (OPKP) in tako omogoči izpis tudi



Slika 5.20: Prikaz receptov v zavihku kulinarika. Vsak recept ima svojo sliko, naslov, ter podatek o zahtevnosti kuhanja in trajanju postopka.

podrobnejših vrednosti. Pod prikazom osnovnih hranilnih vrednosti, je tudi povezava “Prikaži več vrednosti”, katera nam odpre seznam vseh podatkov, ki so na voljo za ta produkt.

Kot predzadnja stran, je stran tipa “Povezave” (Slika 5.22) in je namenjena dodajanju informacij in povezave na druge spletne strani. Posamezne povezave so grupirane v skupine z naslovom, pod vsakim naslovom pa so pripadajoče povezave. S takim grupiranjem smo dosegli boljši pregled na vseh povezavami, katere so vključene na spletno stran. Povezava lahko preusmeri na spletno stran, kakor tudi na datoteko.

Zadnji tip strani pa je tipa “Kontakt” (Slika 5.23) in je namenjena zunanjim obiskovalcem, da kontaktirajo avtorje strani. Namenjeno je predvsem kontaktom z željo po vključitvi h projektu, lahko pa so povpraševanja samo informativne narave. Na desni strani je pozdravno besedilo z nagovorom, zakaj nas kontaktirati. Na levi strani pa je spletni obrazec, z več različnimi polji, katere obiskovalec izpolni, po izpolnitvi, pa obrazec lahko odda s klikom



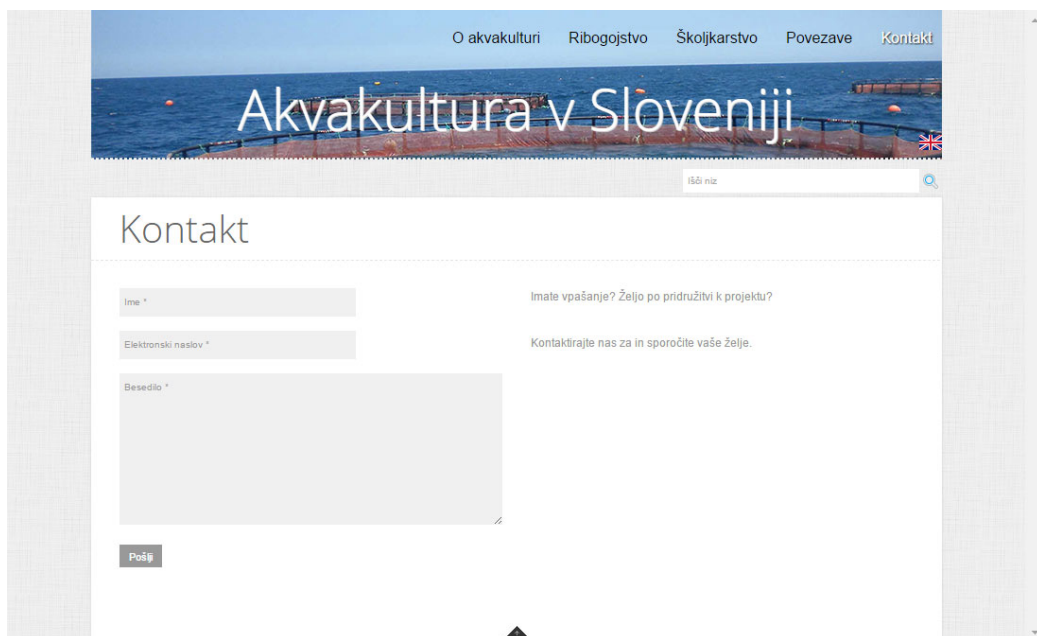
Slika 5.21: Stran za predstavitev produkta Sveži Fonda Brancin neočiščen.

na gumb “Pošlji”. V kolikor kakšnega polja nismo izpolnili, nas stran opozori in prikaže rdeč trikoten znak pred poljem, kateri nakazuje napako v polju. Ob uspešni oddaji obrazca, se namesto obrazca izpiše zahvalno besedilo, v ozadju, pa se oddani podatki vpišejo v podatkovno bazo, prav tako pa se tudi pošljejo v obliki elektronskega sporočila vsem prejemnikom, določenim v administrativnem vmesniku.





Slika 5.22: Stran s povezavami je prikazana po skupinah, znotraj katerih so posamezne povezave.



Slika 5.23: Stran za pošiljanje sporočil z vprašanji ali komentarji.

## Poglavje 6

# Sklepne ugotovitve

Spletni portal smo s pomočjo HTML, CSS verzije 3 in JavaScript dodatka Isotope razvili tako, da se prilagaja tako mobilnim napravam, tabličnim računalnikom, kot tudi namiznim računalnikom, ter tako z eno rešitvijo zagotovili prikaz na vseh tipih naprav. S programsko opremo SilverStripe smo realizirali načrtovan podatkovni model in izpis podatkov. Z dodatnimi skupinami uporabnikov, smo zagotovili dostop do urejanja vsebin zunanjim uporabnikom, kot so uredniki portala akvakulture, ter partnerji.

Partnerjem, ki so vključeni v projekt smo zagotovili svoj predstavitevni del na portalu slovenske akvakulture in jim tako omogočili, da se predstavijo na spletu. Partner ima na voljo predstavitevno stran, stran z zemljevidom, predstavitev produktov, ter možnost vnosa poljubnih vsebin. Podjetje se tako predstavi v slovenski in angleški različici spletnega portala. Tako želimo povzdigniti razpoznavnost akvakulture in osveščenost ljudi, k lokalni podpori te panoge.





# Literatura

- [1] SilverStripe [Online]. Dosegljivo:  
<https://www.silverstripe.org/> [Dostopano 30. 10. 2015]
- [2] Kaj je PHP [Online]. Dosegljivo:  
<http://php.net/manual/en/getting-started.php> [Dostopano 30. 10. 2015]
- [3] Vse o MySQL [Online]. Dosegljivo:  
<http://www.mysql.com/> [Dostopano 30. 10. 2015]
- [4] O CSS [Online]. Dosegljivo:  
[http://www.w3schools.com/css/css\\_intro.asp](http://www.w3schools.com/css/css_intro.asp) [Dostopano 1. 11. 2015]
- [5] O SQL [Online]. Dosegljivo:  
[http://www.w3schools.com/sql/sql\\_intro.asp](http://www.w3schools.com/sql/sql_intro.asp) [Dostopano 15. 1. 2016]
- [6] O HTML [Online]. Dosegljivo:  
[http://www.w3schools.com/html/html\\_intro.asp](http://www.w3schools.com/html/html_intro.asp) [Dostopano 1. 11. 2015]
- [7] Domača stran W3C [Online]. Dosegljivo:  
<https://www.w3.org/> [Dostopano 26. 2. 2016]
- [8] Akvakultura, Ministrstvo za kmetijstvo, gozdarstvo in prehrano [Online]. Dosegljivo:  
[http://www.mkgp.gov.si/si/delovna\\_podrocja/ribistvo/akvakultura/](http://www.mkgp.gov.si/si/delovna_podrocja/ribistvo/akvakultura/)  
[Dostopano 8. 1. 2016]

- [9] Akvakultura, ribiški sklad [Online]. Dosegljivo:  
<http://www.ribiski-sklad.si/Akvakultura> [Dostopano 8. 1. 2016]